IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 19, 2020, accepted February 5, 2020, date of publication February 11, 2020, date of current version February 20, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2973260

Ingress of Threshold Voltage-Triggered Hardware Trojan in the Modern FPGA Fabric–Detection Methodology and Mitigation

SOHAIB ASLAM, IAN K. JENNIONS, MOHAMMAD SAMIE, SURESH PERINPANAYAGAM, AND YISEN FANG

Integrated Vehicle Health Management (IVHM) Centre, Cranfield University, Cranfield MK43 0AL, U.K.

Corresponding author: Sohaib Aslam (s.aslam@cranfield.ac.uk)

ABSTRACT The ageing phenomenon of negative bias temperature instability (NBTI) continues to challenge the dynamic thermal management of modern FPGAs. Increased transistor density leads to thermal accumulation and propagates higher and non-uniform temperature variations across the FPGA. This aggravates the impact of NBTI on key PMOS transistor parameters such as threshold voltage and drain current. Where it ages the transistors, with a successive reduction in FPGA lifetime and reliability, it also challenges its security. The ingress of threshold voltage-triggered hardware Trojan, a stealthy and malicious electronic circuit, in the modern FPGA, is one such potential threat that could exploit NBTI and severely affect its performance. The development of an effective and efficient countermeasure against it is, therefore, highly critical. Accordingly, we present a comprehensive FPGA security scheme, comprising novel elements of hardware Trojan infection, detection, and mitigation, to protect FPGA applications against the hardware Trojan. Built around the threat model of a naval warship's integrated self-protection system (ISPS), we propose a threshold voltage-triggered hardware Trojan that operates in a threshold voltage region of 0.45V to 0.998V, consuming ultra-low power (10.5nW), and remaining stealthy with an area overhead as low as 1.5% for a 28 nm technology node. The hardware Trojan detection sub-scheme provides a unique lightweight threshold voltage-aware sensor with a detection sensitivity of 0.251mV/nA. With fixed and dynamic ring oscillator-based sensor segments, the precise measurement of frequency and delay variations in response to shifts in the threshold voltage of a PMOS transistor is also proposed. Finally, the FPGA security scheme is reinforced with an online transistor dynamic scaling (OTDS) to mitigate the impact of hardware Trojan through run-time tolerant circuitry capable of identifying critical gates with worst-case drain current degradation.

INDEX TERMS Ageing mechanism, field programmable gate array (FPGA), hardware Trojan, negative bias temperature instability (NBTI), propagation delay, reliability, threshold voltage.

I. INTRODUCTION

A modern FPGA is not merely an emulator but a hardware accelerator with heterogenous hard IP cores, such as complex memory blocks, multiple processors, and DSP blocks. Systems on chip (SoC), network on chip (NoC), and adaptive compute acceleration platform (ACAP) are the significant performance and functional enhancements of FPGAs, that have been made possible due to the continual shrinking of transistor sizes down to the scales of 10 nm and below. The performance benefits, however, are limited by power and timing closures. Similarly, the geometric structures of FPGAs with much less silicon and relatively more oxide and moulding compound complicate the heat conduction paths [1]. On the one hand, where it may deteriorate the worst-case heat dissipation route, a given power density, on the other hand, produces a significant temperature variability [2]. This results in a higher temperature for the same amount of power dissipation. It is, therefore, essential to consider thermal variation as an on-going challenge for advanced technology nodes alongside the associated issues of power and timing closures.

Looking at the FPGA fabric, we find a mesh of layers comprising a substrate, high-k dielectric interfaces, and metal

The associate editor coordinating the review of this manuscript and approving it for publication was Luis Javier Garcia Villalba.

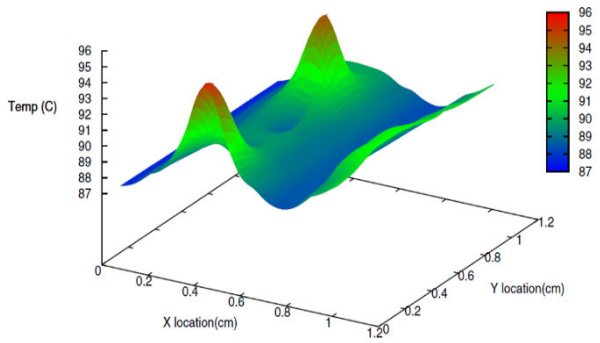


FIGURE 1. Thermal profile depicting hotspots in an FPGA [4].

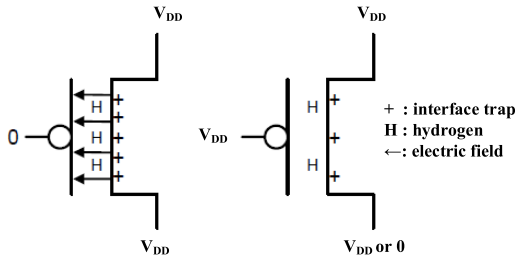


FIGURE 2. NBTI mechanism in a PMOS transistor.

interconnects. Each layer has a varying range of thermal conductivity with silicon dioxide sitting at 1.3-0.3W/mK, and copper metal interconnects going as high as 400 W/mK [3]. These differences in thermal conductivity affect the heat transfer and introduce variations in temperature across the FPGA area, thereby creating hotspots as can be seen in Fig. 1. The resultant increase in temperature and appearance of hotspots across the FPGA surface causes non-negligible variations in the timing and power domains of the design [4]. This non-uniform thermal dissipation aggravates the ageing mechanism of negative bias temperature instability (NBTI) and leads to accelerated ageing of the FPGA fabric.

The NBTI ageing mechanism is dominated by a negative shift in threshold voltage (V_{th}) of pMOSFETs that make up the FPGA, along with nMOSFETs. The change in threshold voltage is in response to biasing in the strong inversion region, which causes the disintegration of Si-H bonds at the oxide interface due to the presence of holes within the pMOS inversion layer, as is evident in Fig. 2. This bond disintegration process creates positively charged interface traps, which, along with new or existing traps within the oxide, increases the threshold voltage [5]–[7].

Undeniably, NBTI is well known to researchers and manufacturers alike as a dominant ageing mechanism in all different configurations of integrated circuits (ICs). For instance, in the post-IC manufacturing period of 7 to 10 years, accelerated ageing due to NBTI has been reported by [5] and [8] as degradation in threshold voltage up to 50 mV. Speed degradation (of 20%) follows these shifts in threshold voltage and, therefore, shows a strong correlation between NBTI prompted delay and threshold voltage shift.

It is important to note that the non-uniformity of NBTI (*due to different thermal conductivity patterns*) across the chip surface affects various blocks within the FPGA differently. As a result, the delay variations induced by NBTI, across the FPGA surface, could potentially generate new critical paths, which, in turn, may prevent an efficient and balanced timing closure [9]. In the case of data paths, for instance, an increase in gate delays causes a late transition of an input signal at the flip-flop. Such varying transitions violate the flip-flop setup and hold time that eventually results in the sampling of flawed values at the output of the data path.

These variations, apart from being the primary source of FPGA reliability concerns, also affect the integrity of logic applications and aggravate to levels that may lead to system failures. More alarming is the **hardware security threat** that can leverage the dwindling reliability of an FPGA device under NBTI influence. It can jeopardise FPGA's optimal performance with the insertion of malicious and stealthy circuitry, called hardware Trojan – designed by exploiting stochastic and systematic variation patterns that exist within the FPGA.

The exacerbation of NBTI, owing to the continual transistor miniaturization, is fast becoming a major donor of the process of ageing in downscaled technology nodes. It poses a challenge for the proponents of high FPGA reliability and performance to understand the dynamics of NBTI in designing a hardware Trojan, initially, from an intruder's (*a rogue element*) perspective and lately by designing a threshold voltage-aware sensor for its detection, followed by an effective mitigation methodology from security assurance and defender's perspective.

In other words, it implies the development of an FPGA security scheme (Fig. 3), which assumes that an intruder is capable of capturing and analysing the shifts in threshold voltage of pMOSFETs (*that result in lowering the frequency, signal path delay variations, and flawed transitions*) due to the NBTI effect. If successful, the intruder may design and insert a stealthy malicious circuit (*called hardware Trojan*) inside the FPGA. With sufficient parametric information and precise monitoring, the intruder may capitalize NBTI ageing mechanism to activate a dormant hardware Trojan. This is further elaborated in the threat model described in section-IIA.

It is well established that the detection of such hardware Trojans is difficult using testing techniques like built-in self-test (BIST) because no test vector can activate an ageing effect [10]. The process of accelerated stress and ageing test on the affected node may, however, reveal such Trojans; however, the process, when performed on a complete integrated circuit, is time and cost-intensive [11].

In this paper, we direct the FPGA security scheme, shown in Fig. 3, towards the design and implementation of a threshold voltage-triggered hardware Trojan in a lower technology node (*28 nm FPGA*). A degradation in the drain current,

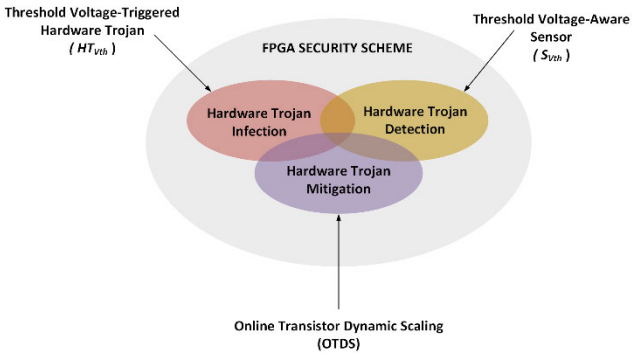


FIGURE 3. FPGA security scheme comprising hardware Trojan Infection, Detection, and Mitigation sub-schemes.

oscillation frequency, and the subsequent increase in the response time (*due to shift in threshold voltage*) of the 28 nm FPGA is observed through a novel sensor. An effort is also made to mitigate the impact of a hardware Trojan by introducing a method of compensation that enhances the current flow and lowers the rise in delay due to NBTI. This includes an online transistor dynamic scaling (OTDS) approach as a mitigation methodology to counter hardware Trojans.

The proposed designs and implementations are verified and validated using post-layout, and Monte Carlo simulations with Cadence Virtuoso ADE tools, followed by real-time experiments on Global Foundries fabricated 28 nm technology node. Threshold voltage-triggered hardware Trojan, '$\boldsymbol{HT_{Vth}}$,' operates in a threshold voltage region of 0.45V-0.998V, consuming ultra-low power (10.5nW), and remaining stealthier within an area overhead of as low as 1.5%. The Threshold Voltage-aware sensor, '$\boldsymbol{S_{Vth}}$,' utilizes 3% of die resources and achieves the detection sensitivity of 0.251mV/nA. OTDS enables the auto-resizing of transistors to mitigate the impact of hardware Trojan payload due to NBTI-based threshold voltage shifts falling between 10% and 90%.

A. CONTRIBUTION

This research work entails some key contributions. Firstly, we have provided a composite solution for security and reliability-threatened FPGAs, named as **FPGA Security Scheme (Fig. 3)**. It involves: 1) Ingress of a stealthy threshold voltage-triggered hardware Trojan-(**HT Infection Scheme**), 2) Detection of hardware Trojan using lightweight Threshold Voltage - aware sensor ($\boldsymbol{S_{Vth}}$)-(**HT Detection Scheme**), and 3) Mitigating the impact of hardware Trojan using online transistor dynamic scaling (***OTDS***)-(**HT Mitigation Scheme**). Secondly, the development of a stealthy hardware Trojan based on a combinatorial and sequential circuitry and NBTI ageing mechanism is one of its kind - operating in a subthreshold region makes it highly sensitive to shifts in threshold voltage and trigger the NBTI ageing mechanism. Thirdly, the lightweight Threshold Voltage-aware sensor is based on a fixed and dynamic pair of ring oscillators, capable of detecting small ageing levels through precise measurement of frequency and corresponding delay variations (*in conformity with shifts in threshold voltage*). And finally, a novel technique for mitigating hardware Trojan impact is proposed that provides a run-time tolerance circuitry capable of identifying critical gates with worst-case current degradation and subsequent transistor re-sizing to revive healthy current values. Equally significant is the fact that these schemes have been developed, keeping in mind the goal of achieving absolute optimization of the area and power overheads.

The rest of the paper is organised as follows. Section II gives information on the related work with a brief critique. In Section III, we delineate the design, simulation, and implementation of Threshold Voltage-triggered hardware Trojan, '$\boldsymbol{HT_{Vth}}$,' in a 28 nm technology node FPGA. Section IV presents the design and implementation of a Threshold Voltage-aware sensor ($\boldsymbol{S_{Vth}}$ - the hardware Trojan Detector) and discusses various options tested to achieve high sensor accuracy. In section-V, the mitigation technique based on online transistor dynamic scaling (auto-resizing) and its correlation with NBTI-induced performance degradation are highlighted. Section-VI puts forth the implementation and optimization of the HT mitigation scheme, whereas Section-VII provides its simplistic comparison with some of the state-of-the-art reliability and security solutions. Section-VIII concludes the paper with a future course of work.

II. RELATED WORK

Extensive research has been undertaken to present a detailed analysis of ageing and performance degradation in integrated circuits. It mainly involves the fingerprinting of ICs' electrical parameters (voltages, currents, frequencies, and EM signals) by retrofitting well-designed on-chip sensors and structures. Be it the detection of counterfeits, recycled ICs, or detection and mitigation of hardware Trojans; the same parameters are manipulated by researchers to understand different undesired behaviour patterns and anomalies in ICs (ASICs, FPGAs, and Microprocessors) for remediation and building effective countermeasures.

In [12], Karhunen Loéve theorem is used to study the power consumption behaviour of hardware Trojan infected FPGA to determine the possibility of its detection. This technique considers the impact of process variations that occur within the FPGA; however, it avoids the noise factor and is limited to simulation analysis. Similarly, the researchers in [13] have again simulated and analysed the occurrence of path delays in the signals of various logic applications using the embedded monitors. Both of these techniques do not provide real-time analysis. An integrated hardware system capable of monitoring the behaviour of critical interconnects (*wires*) is proposed in [14]; however, it does not provide sufficient information on the efficiency of this method. In [15], a test methodology to ease hardware Trojan triggering by increasing its electrical activity is proposed for early detection. In [16], an attempt to carry out precise measurement of an IC's operating frequency, maximum frequency (f_{max}),

and its dynamic power consumption is made by lowering the impact of process variations. However, the calculation of the accurate value of f_{max} is quite challenging and also susceptible to 'false positives.'

The use of ring oscillators' sensitivity to variations in temperature and power enables the detection of medium-to-heavyweight hardware Trojans, however, not effective against the small-sized/lightweight hardware Trojans [17]. The researchers in [18] have created a network of ring oscillators spread across the FPGA surface to capture the changes in their oscillation frequency due to the presence of hardware Trojan. This is validated using a digital storage oscilloscope (DSO) and later analysed using the principal component analysis to differentiate between the genuine and the HT infected FPGA. However, when applied to an ASIC [19], this technique suffers from the lower levels of measurement accuracy due to the usage of an 8-bit counter instead of a digital storage oscilloscope, questioning the accuracy of on-chip designs.

In [20], the clustering methodology is proposed, whereby dedicated sensors are embedded in the power grids of different voltage islands in FPGA, to enhance HT detectability. However, it does not provide adequate experimental evidence to evaluate the efficacy of this methodology. The capturing of electromagnetic signatures of target applications in ICs has also been studied for hardware Trojan and anomaly detection. For instance, a method based on electromagnetic (EM) cartography is proposed in [21], but then again, due to inappropriate method of interpretation of EM traces, the detection of hardware Trojans remains low. On the other hand, in [22], the researchers have devised an improved technique that interprets the EM traces optimally. By controlling and maintaining the temperature during EM measurements, this technique improves the probability of detecting lightweight hardware Trojans. Further to this, the researchers in [23] are able to differentiate between the healthy and HT infected population of FPGAs through a comprehensive analysis of EM signatures.

A reasonable amount of work has also been undertaken to design and develop various sensing techniques and frameworks for the detection and mitigation of the NBTI mechanism and its noticeable impact. In [24], an analog supply-devoid NBTI sensor is proposed to eliminate noise; however, the input of other external signals makes its operation very complicated during the stress and recovery as well as measurement modes. This reduces its overall measurement accuracy. In [25], the dynamic reliability of the device is managed using NBTI and HCI (Hot Carrier Injection) sensors. In this case, the threshold voltage of the stressed device is measured and transformed into the delay function. However, these sensors are less sensitive to temperature variations and occupy large device area with high power consumption. In another study () [26] an NBTI sensor is designed to measure the standby leakage current ($\boldsymbol{I_{ddq}}$). Designed explicitly for SRAM cells, this sensor monitors the leakage current, characterising temporal degradations. It, however, requires an additional bias generator to maintain active load on the sensor, which results in non-linearity and reduced sensitivity to the input signal. Researchers in [27] have used the current-mirroring technique to capture NBTI based degradation. The power supply current is mirrored and subsequently transformed into voltage. The drawback of this approach lies in the usage of power gating that slows down the response time of the sensor. However, its performance is relatively more stable than the $\boldsymbol{I_{ddq}}$ based sensor.

To mitigate NBTI ageing and degradation impact on the reliability and performance of an IC, we have come across the concept of one-time design constraints put forth by various researchers. For instance, [28], [29] suggest an increase in supply voltage to manage and control NBTI. This may, however, lead to power and thermal overheads – an undesirable design feature. Whereas [30] and [31] propose transistor oversizing and reduction in the clock frequency, respectively as an optimum NBTI mitigation. The thermal management of ICs via different cooling arrangements is also proposed to contain and reverse the NBTI impact [4]. Gate replacement technique is proposed in [32] that attempts to optimize the NBTI ageing effect. Techniques on the balancing and removal of stress to control short-duration threshold voltage instability are suggested by Choi et al [33]. These, however, fail to consider the critical factor of prolonged ageing effect at high temperatures. In [34], Kiamehr et al. have highlighted the use of ageing-aware library standard cells to mitigate BTI impact on the rise and fall times of different signals. The threshold voltage shift is, initially, measured and later used to optimize the width ratio (W_p/W_n) of each transistor to counter the ageing effect. However, its applicability for IC run-time is not considered. Another study by Zhang et al. [35] describes the techniques that involve the identification of critical gates and their replacement with NBTI-tolerant gates. The use of dynamic voltage scaling and data flipping has also been proposed by [36] to recover the static noise margin in the case of SRAMs.

The measurement of a beat frequency between the reference and stressed ring oscillators using a silicon odometer is also proposed in [37] to keep track of degradation due to NBTI. Similarly, a hybrid scheme comprising ring oscillators and delay line based online-ageing monitoring is presented in [38] for the measurement of degradation. These sensor schemes are, however, focused on ensuring precise measurements rather than triggering accelerated degradation to detect the presence of any notable anomaly. In order to fill in this gap, a low-cost and lightweight structure consisting of ring oscillator based sensors for in-field capturing of IC/FPGA ageing is proposed in [39] to enhance the granularity of detection.

More recently, authors in [40] have proposed a multitype hardware Trojan protection framework, called RG-Secure. This framework is designed and validated to provide RTL and gate-level security to FPGA based SoCs (*deployed in IoT environment*) against different types of hardware Trojans by merging 3PIP (third party intellectual property) trusted

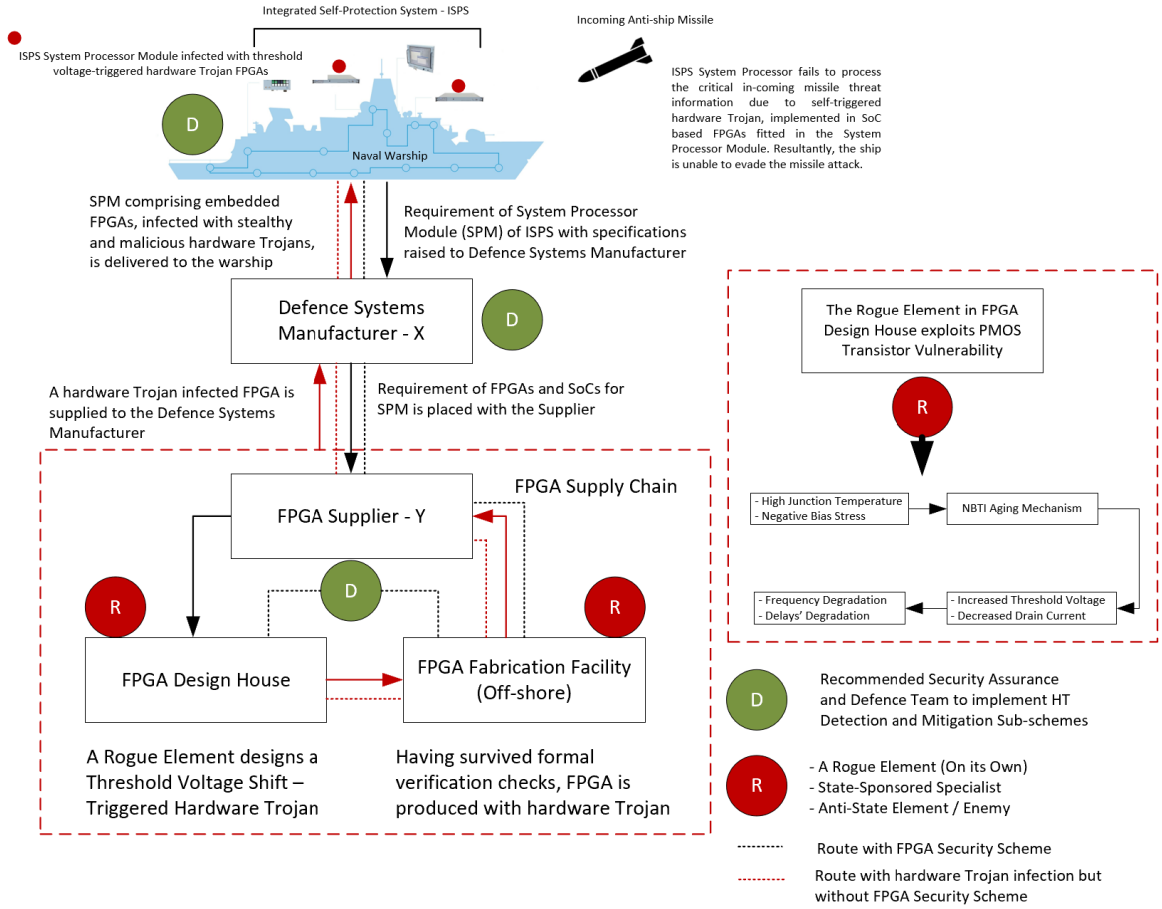


FIGURE 4. **Threat Model: A novel self-triggered Threshold Voltage-Shift based Hardware Trojan 'HT_{Vth}' is designed and implemented by a rogue element in a 28 nm FPGA used in System Processor Module of ISPS (Integrated Self Protection System) of a Naval Warship.**

design approaches with the scan-chain netlist feature analysis. Employing tree-based learning algorithms, they have shown a good hardware Trojan detection coverage at RTL and gate-levels, with 100% true positive rate and 94% true negative rate accuracies. In our opinion and analysis, this method/framework holds true for less complex netlist structures and scan-chain features. However, it may not be effective against parametric hardware Trojans (*e.g., threshold voltage-triggered*) that have netlists of distinct structure and trigger behaviour.

Our work, however, follows an integrated approach, as mentioned earlier, and encompasses three elements namely, HT insertion (*infection*), its detection, and mitigation. We build these elements considering the limitations and strengths of the abovementioned techniques and different on-chip sensors' architectures, with FPGA security and reliability in perspective.

A. THREAT MODEL

Hardware Trojan, a stealthily malicious entity, capable of inflicting performance degradation, sensitive information disclosure, and functional disorder at the micro-architectural level in FPGAs, continues to challenge the efforts toward strengthening hardware security. In an attempt to control its increasing threat, we construct a threat scenario/model to understand its implications for a high-end defence asset - a naval warship, fitted with an 'Integrated Self-Protection System' (ISPS) and eventually develop a full-spectrum FPGA security scheme.

ISPS is a real-time functional integration of electronic warfare systems used onboard naval warships and fighter aircraft as well. It comprises Electronic Support Measures' (ESM) systems like Radar Warning Receivers (RWR), System Processor for threat environment assessment and asset assignment, and Electronic Counter Measures' (ECMs) systems like Jammers and Chaff launchers.

We, however, focus on System Processor Module and regard it as a vulnerable entity in ISPS system due to its high probability of infection with security-compromised FPGAs. The threat scenario, as depicted in Fig. 4, has three main elements, namely: 1) the naval warship, 2) the Defence Systems Manufacturer - X, and 3) the FPGA Supplier - Y. The red sphere with letter 'R' represents the 'Rogue Element' that could be working with malicious intentions on its own, as a state-sponsored VLSI design specialist, or an anti-state element/enemy. We assume its presence at FPGA Supplier

premises in 'Design House,' 'Fabrication Facility,' and 'SoC Integration Section' - all representative of the FPGA supply chain. The green sphere with the letter 'D' represents the authors' recommendation on forming a 'Security Assurance and Defence Team' to counter the malicious insertion in FPGA by the rogue element. Its presence is recommended in all three elements.

The threat process begins with the naval warship placing the requirement of a new System Processor Module (*installed with n-number of FPGAs, providing vital electronic warfare functions*) for the ISPS system from the Defence Systems' Manufacturer-X. Subsequently, the FPGA supplier -Y is subcontracted by X to provide FPGAs built on 28 nm process technology. A rogue element R, stationed in a Y design house, receives the task of designing the FPGA. Here, we assume that R is an expert FPGA designer with sufficient working knowledge of FPGA design flow, specific to the insertion of stealthy hardware Trojan based on the threshold voltage shifts in PMOS transistors. Such type of hardware Trojans corresponds to the functionality level parametric characterization [41] and are targeted at paralysing device/system functionality. To maintain undetectability, R employs '**Split hardware Trojan Insertion**' methodology, whereby a part of a hardware Trojan circuit is built at the design stage in the design house. Post design and successful simulation, the design file (GDSII) is forwarded to the FPGA fabrication facility for manufacturing. Here, the remaining part of hardware Trojan is added (*at the RTL and Gate level*) post-manufacturing reliability tests by another rogue element (*collaborator*) at the FPGA fabrication facility to evade detection. As per our recommendation (mentioned in Fig. 4), if D is also stationed at the design house, it will design detection and mitigation circuitry in addition to the hardware Trojan circuit design by R (*with both D and R remaining oblivious of each other's work*). The newly fabricated chips are now ready for installation on the system processor module at X. The security assurance and defence team D at X carries out pre-installation security tests to observe anomalies specific to hardware Trojan based on threshold voltage shifts. If the tests are clear, the FPGA is installed on the system processor module and delivered to the end-user - the naval warship. At this point, we make two assumptions. Firstly, if the detection and mitigation circuitry fails and the hardware Trojan gets triggered, the damage to ISPS operation ability will occur. Secondly, if the detection and mitigation circuitry successfully detects and mitigates the hardware Trojan, the ISPS system will continue performing efficiently without any hindrances, provided some other faults that are not related to hardware Trojan erupt. As can be seen in Fig. 4, we have also recommended the placement of D in the naval warship. So, before installing the system processor module in the ISPS system for harbour and sea acceptance trials (HATs and SATs), D must carry out security tests to challenge the first assumption and in case of it holding true, return the module to X for replacement.

In a nutshell, as shown in Fig. 4, if the 'red-dotted line' route (*containing the FPGA infected with hardware Trojan but without any detection and mitigation component of FPGA security scheme*) is adopted, the hardware Trojan would remain undetected and get triggered with pre-defined threshold voltage shift, thereby causing ISPS system performance degradation and leaving the ship vulnerable to a devastating missile attack. On the other hand, if the 'black-dotted line' route (*containing a robust FPGA security scheme*) is assumed, the hardware Trojan can be easily detected and denied triggering, thereby keeping the ISPS system proficient in thwarting any external threat to the ship.

Considering the above threat scenario/model, we, in the following sections, make an effort to sequentially develop a realistic FPGA security scheme for the security assurance and defence team to not only provide security and dependable redundancy to critical systems like ISPS but also augment the post-manufacturing tests regime (*security tests, in specific*) employed by FPGA manufacturers. The first step, in this regard, is the design and implementation of a hardware Trojan itself, followed by detection and mitigation circuitries based on the Trojan's impact on target FPGA applications.

III. THRESHOLD VOLTAGE-TRIGGERED HARDWARE TROJAN

In line with the FPGA security scheme (Fig. 3), we define the contours of the hardware Trojan (HT)-infection scheme. It encompasses an operational system's FPGA (28 nm technology) vulnerable to ingress of hardware Trojan, which in turn, inflicts operational and functional damages to the system and its various components.

Beginning with HT-infection scheme, we construct a hardware Trojan with details as follows:

A. DESIGN CONSIDERATIONS

As mentioned earlier, the high temperature activates the NBTI mechanism in the FPGA silicon fabric. Resultantly, it accelerates the process of ageing and leads to undesirable characteristics. For instance, temperature changes beyond 75° C between different layers of a substrate could cause variations in interconnect delays up to 31-38% [42]. Subsequently, the device tends to operate slower with delays also observable in the control and data signals. Such timing inconsistencies cause synchronous circuits transit into redundant states or momentary glitches. However, to avoid failures, the clock period can be managed to counter the system glitches. The authors in [43] have, nevertheless, suggested that despite clock management, the period of momentary glitches tends to increase with NBTI and may set off pre-determined activity related to malicious circuitry.

Tabular analysis (Table 1) of the results obtained by [9] reveals that:(a) the shift in threshold voltage (V_{th}) and drain current (I_{dd}) is a function of high temperature and is observed to increase for V_{th} and decrease for I_{dd} at temperatures $\geq$ 60°C, (b) an approximate rise of 4% in the threshold voltage shift is evident with the scaling down of technology nodes [44]. The rate of decrease in I_{dd} is, however, less than the rate of V_{th} increase, and (c) eventually, the propagation

TABLE 1. Impact of NBTI aging mechanism on PMOS transistor parameters.

Technology Node	Temp. (°C)	V_{th} Shift (mV)	I_{dd} Degradation (%)	Delay Degradation (%)
90 nm	25	17.68	3.42	4.01
	75	21.43	4.82	5.73
	125	23.96	5.22	6.63
65 nm	25	18.22	4.83	5.82
	75	23.12	6.20	8.06
	125	25.74	6.86	9.20
45 nm	25	20.68	5.03	8.75
	75	25.02	7.50	9.64
	125	29.81	7.85	11.51
32 nm	25	21.05	5.89	9.25
	75	26.25	8.25	10.50
	125	32.55	9.76	13.82
28 nm	25	21.55	6.09	9.83
	75	26.91	8.92	11.25
	125	33.15	10.32	14.49

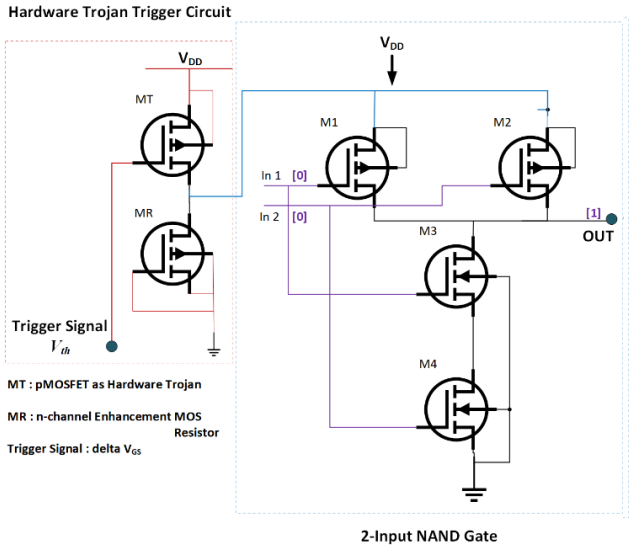


FIGURE 5. Schematic of a threshold voltage-triggered hardware Trojan (HT_{Vth}) in a combinatorial circuit (2-input NAND gate).

delays increase with the aforementioned trends of variation in V_{th} and I_{dd}.

In light of the above, the essential design targets for threshold voltage-triggered hardware Trojan (HT_{Vth}) are set accordingly such that: (a) the transfer function of the Trojan circuit must be linear. (b) sensitivity to temperature and threshold voltage changes should be significantly high, (c) the change in the output should be significantly high for a change in the input, and (d) negligible temporal degradation and tolerance to process variations should be maintained.

Additionally, the element of stealthiness and undetectability of hardware Trojan is highly significant (*primarily from the perspective of a rogue element*). Hardware Trojan, by definition, has to be stealthy to escape detection. In order to achieve this, we have ensured during design and implementation stages (*described in the following sections*) that the size of the circuitry is as small as possible with equally low power consumption and without compromising the effectiveness of its payload. Regarding the area and resource utilization at the circuit and RTL/Gate level, we have used as minimum instantiation as possible to ensure low area and power overheads. These have been measured to be at just **1.5 %** of the total available resources on a 28 nm process technology. With such a small percentage, it is highly unlikely that the added circuitry of hardware Trojan would be discovered either during post-manufacturing tests or during run-time monitoring. Hence in a multi-million gates chip, it can hide easily. Also, more importantly, the proposed threshold voltage triggered Trojan does not draw any extra current while dormant; therefore, it becomes challenging even to detect it through power signature analysis.

B. ARCHITECTURE OF THRESHOLD VOLTAGE TRIGGERED HARDWARE TROJAN (HT_{VTH})

We propose a circuit implementation of threshold voltage-triggered hardware Trojan, HT_{Vth}, which is valid for CMOS devices. The implementation is demonstrated for both the sequential and combinatorial logic as follows:

1) CONCEPTUALISING HARDWARE TROJAN IN COMBINATORIAL CIRCUITS

Considering the **combinatorial** circuit for hardware Trojan, a 2-input NAND gate is designed to have two PMOS transistors M1 and M2 parallel to one another. These are then connected in series to two NMOS transistors M3 and M4, as shown in Fig. 5. The drain terminals of both M1 and M2 are shared and connected to the source terminal of M3. The output of the NAND gate is tapped out at M3. Another PMOS transistor, MT (Trojan Transistor), is constructed in series with a MOS resistor (MR) to work as a hardware Trojan. The MOS resistor acts as a current limiter as soon as the triggering signal is received at the MT gate terminal. A compact silicon area of $\mathbf{50}\mu\mathbf{m}^{\mathbf{2}}$ is occupied by this circuitry with a low power consumption of $\mathbf{1.05}\ \mu\mathbf{W}$.

Operationally, the Trojan is kept in the 'ON' stealthy state so that the transistors M1 and M2 remain connected to the power supply (V_{DD}). The output of the NAND gate, on the other hand, is '0' when both of its inputs A and B are '1'. Otherwise, the output always remains at '1'. As MT, the hardware Trojan receives an NBTI induced shift in threshold voltage (triggering signal) at its gate terminal; it initiates the process of accelerated device ageing with elevated temperatures and reduced frequency of the NAND gate circuitry. The shift in the threshold voltage, which acts as a trigger for the hardware Trojan, needs to be measured very carefully. For this purpose, we have also designed a threshold voltage measuring circuit, termed as '**Threshold Voltage Meter**' (*The detailed configuration of this circuit is given in section IIIC)*. With the value of threshold voltage (V_{th}) exceeding the pre-defined level (***pre-Trojan Trigger Threshold Voltage-*** V_{th_ptt}), a triggering

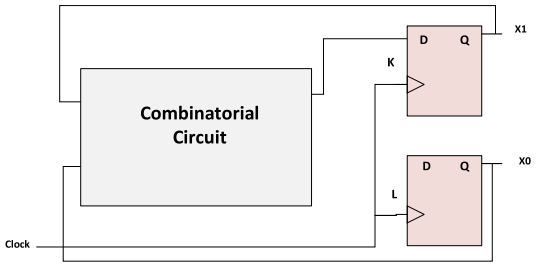


FIGURE 6. Block diagram representation of a sequential circuit.

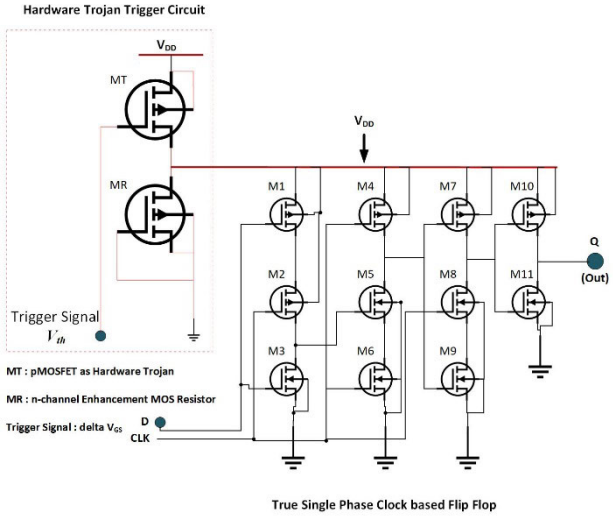


FIGURE 7. Schematic of threshold voltage-triggered hardware Trojan in a Sequential Circuit (TSPC based Flip Flop).

signal is generated at the gate terminal of MT. This active high triggering signal switches the MT 'OFF' and leaves the PMOS transistors M1 and M2 without power, thereby affecting the operation of the NAND gate.

2) CONCEPTUALISING HARDWARE TROJAN IN SEQUENTIAL CIRCUITS

In order to build a **sequential** circuit for hardware Trojan demonstration, we consider adding two flip flops (K and L) to the combinatorial circuit, as shown in Fig. 6 and Fig. 7. The binary decoding with two bits X and Y as the most significant bit (MSB) and the least significant bit (LSB), respectively, are used for the flip flops. An inactive hardware Trojan, MT, is embedded into the flip flop K ***(overall area of this circuitry raises to 75μm^2, consuming power of 1.25μW)***. Under no-triggering and normal operating conditions, the sequential circuit functions optimally without any effect on the dynamic power consumption. As the MT is triggered, the supply voltage ($\boldsymbol{V_{DD}}$) feeding the flip flop 'K' is cut off, resulting in the malfunction of the flow of finite state machine (FSM). Although the flip flop 'L' remains unaffected and healthy, the failing of flip flop 'K' reduces the FSM states to only two high impedance states - z_0 and z_1.

The above structure is further elaborated by constructing a true single-phase clock (TSPC) based flip flop. The payload is the same PMOS transistor MT with a MOS resistor (MR) connected in series to it, as shown in Fig. 5. MT, acting as a switch, controls the connection of the body and source of all PMOS transistors (M1, M2, M4, M7, and M10) in the flip flop. The bodies of all NMOS transistors (M3, M5, M6, M8, M9, and M11) are grounded permanently. When the switch MT is 'ON,' all the PMOS transistors remain connected to $\boldsymbol{V_{DD}}$. On the contrary, when the switch MT is in 'OFF' state, the body and the source of all PMOS transistors are shorted to ground through the resistor, leaving the flip flop without power supply and resulting in circuit malfunction. Similar to the triggering of MT in the combinatorial circuit, the shift in threshold voltage due to NBTI is designed to initiate MT triggering here in the sequential structure as well. A Global Foundries 28 nm process technology is used to accomplish circuit implementations and subsequent logic applications.

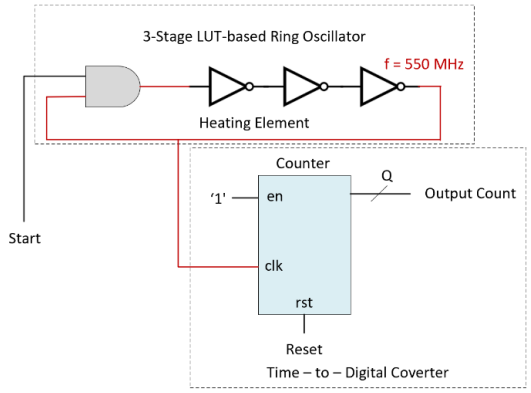
3) ADDING RING OSCILLATOR BASED HEATING ELEMENT FOR ACCELERATED NBTI IMPACT

To accelerate the NBTI ageing mechanism and observe a corresponding shift in threshold voltage '$\boldsymbol{V_{th}}$,' we designed and implemented a LUT-based ring oscillator to act as a heating element for raising the temperature high enough to trigger NBTI. The architecture of the heating element is shown in Fig. 8(a). It is important to note here that this heating element is designed and implemented as an integral part of the hardware Trojan infection scheme.

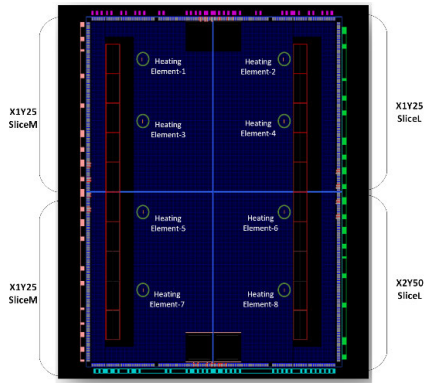
As stated earlier, there exists a strong correlation between the shift in threshold voltage and the die temperature. Taking this into account, a set of eight controllable ring oscillators (ROs), comprising 3-inverter stages and a time-to-digital converter (TDC) each, are implemented across the FPGA fabric (*28 nm technology node*) at locations shown in Fig. 8(b) using the Vivado design suite. It is noteworthy that the number of stages in a ring oscillator determines the toggling frequency and hence, the corresponding amount of heat generation, measurable as a variation in temperature [45]. In order to disrupt the ISPS system, the toggling frequency of an RO must be high enough to generate a large amount of heat per micron for high temperatures. Accordingly, only a single LUT is used to implement RO with 3-inverter stages and a TDC.

We define the area-constraint for our heating elements to only 8 LUTs (0.00025%) out of the total 32,000 LUTs constituting the CLBs. The built-in system monitor is then programmed to access XADC sensor readings of the thermal diode in FPGA. The heating element is enabled/disabled by a time-driven program running on the FPGA, which also keeps reading the temperature values and transmitting them to the workstation via the JTAG interface.

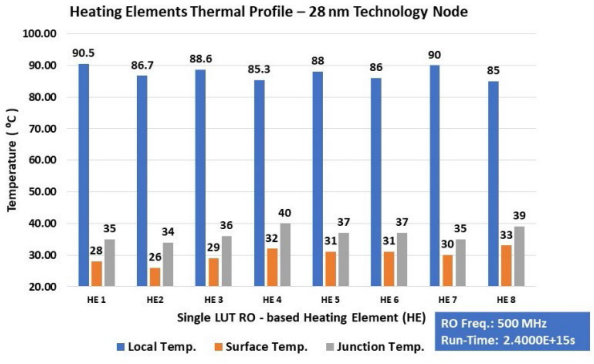
The execution of the experiment is organized in such a way that the die temperature of the FPGA is allowed to stabilise for a period of 35 minutes before enabling the heating element for a period of 40 minutes. Upon completion of this operational phase, the heating element is disabled and allowed to rest for 35 minutes. During this period, the fall in temperature is observed to assess the behaviour of the heating element.



(a)



(b)



(c)

FIGURE 8. **(a) Schematic of a 3-stage ring oscillator-based heating element with Time-to-Digital converter. (b) 28 nm technology node floor-planned with 08 x heating elements. (c) Thermal profile of FPGA (28 nm technology node) with 08 x heating elements.**

Finally, the heating element is again enabled for another 40 minutes to affirm the repeatability and validity of the experiment.

We tested the LUT based ring oscillators (the heating elements) spread over eight different locations on the FPGA as per the procedure mentioned in the previous paragraph and measured it toggling at 550 MHz. The temperature

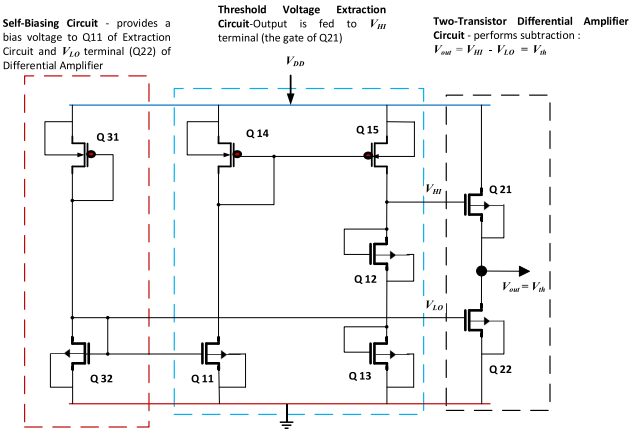


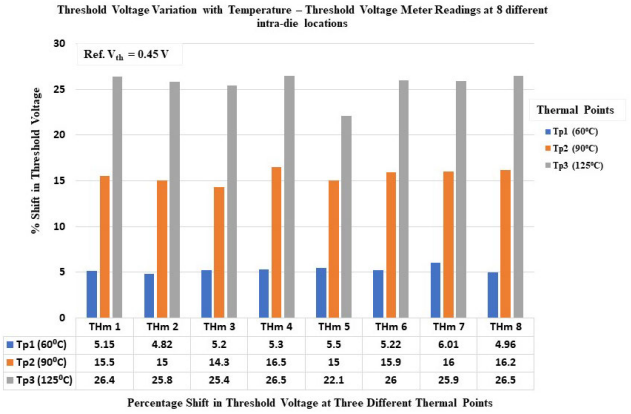
FIGURE 9. **Schematic of threshold voltage meter. The output of the differential amplifier is the threshold voltage (V_{th}).**

measurements were made using the FPGA's internal thermal diode (*for the whole FPGA*), on-chip thermal sensors (*the LUT based RO connected to the counter for local temperature*), and the external laser-based IR temperature gun, positioned over the FPGA package.

Initially, the temperature is stabilised to an idle FPGA state, meaning when it is powered up and configured, with the negligible workload, and without the heating elements enabled. The idle temperature for the whole die (*junction temperature*) is measured to be **10.5°C**, the local RO **10°C**, and the surface **5°C**. The heating elements are subsequently enabled with clock disabled to achieve asynchronous behaviour of LUT based RO and toggle as fast as physically possible without any clock constraint. Upon enabling the heating elements one by one for a period of 40 minutes each, the local, junction, and surface temperatures depicting the thermal profile of an FPGA is obtained, as shown in Fig. 8 (c). It can be seen that the temperatures rise considerably higher to cause shifts in the threshold voltage and accelerate the NBTI degradation mechanism. The threshold voltage meter, described later, continuously measures the voltage till the time the hardware Trojan circuit is triggered at a value above the nominal 'V_{th}' value (0.45V).

C. THRESHOLD VOLTAGE METER

As mentioned earlier, the shift in threshold voltage 'V_{th}' is the manifestation of the ageing mechanism of NBTI in PMOS transistors that make up the FPGA fabric and its primitives. Therefore, the precise measurement of 'V_{th}' is critical for triggering the threshold voltage based hardware Trojan. Accordingly, we design and implement a threshold voltage meter that directly generates an output voltage 'V_{out},' equal to 'V_{th}.' Figure 9 shows the schematic diagram of the meter. As is evident, this circuit has no reference voltage 'V_{ref}' input and is, therefore, a 3-terminal circuit. The transistors Q31 and Q32 provide a bias voltage at the gate of Q11; this voltage is then applied to the low voltage 'V_{LO}' terminal of the differential amplifier, i.e., at the gate of Q22. Whereas, the transistors Q11-Q15 implement a circuit whose output is



	THm 1	THm 2	THm 3	THm 4	THm 5	THm 6	THm 7	THm 8
Tp1 (60°C)	5.15	4.82	5.2	5.3	5.5	5.22	6.01	4.96
Tp2 (90°C)	15.5	15	14.3	16.5	15	15.9	16	16.2
Tp3 (125°C)	26.4	25.8	25.4	26.5	22.1	26	25.9	26.5

FIGURE 10. % Shift in threshold voltage with rise in temperature across 8 different intra-die locations. Threshold voltage meter is used to read V_{th}. Reference V_{th} is pre-defined at 0.45V.

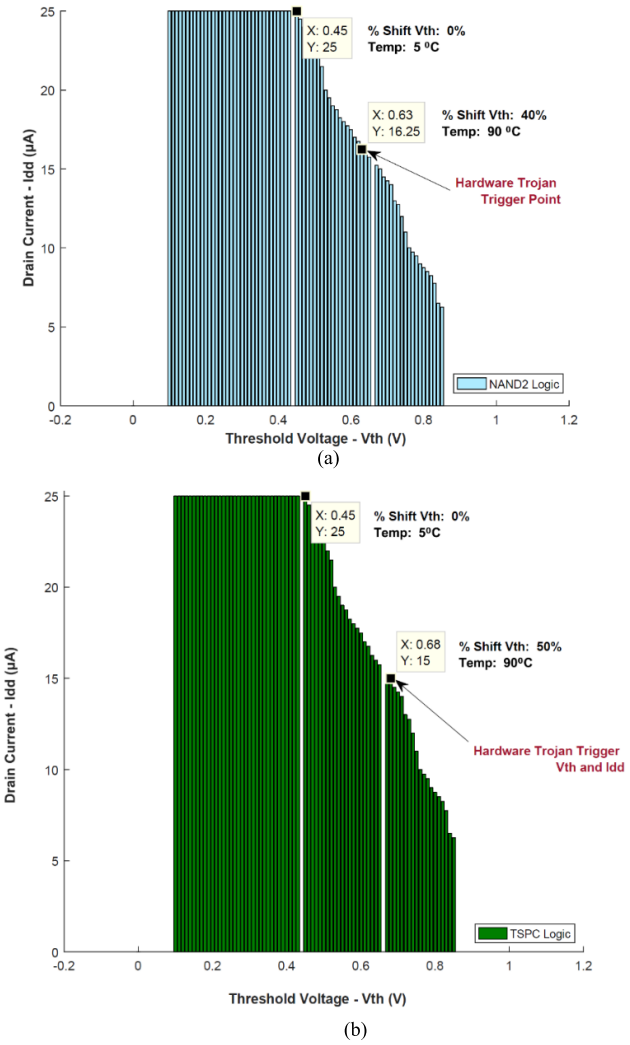


FIGURE 11. (a) An increase of 40% shift in threshold voltage at 90°C degrades the drain current by 35%, triggers the hardware Trojan and impairs the NAND2 logic. (b) An increase of 50% shift in threshold voltage at 90°C degrades the drain current by 40%, triggers the hardware Trojan and impairs the TSPC logic.

applied to the high voltage 'V_{HI}' terminal of the differential amplifier at the gate of Q21. Eventually, the differential amplifier comprising Q21 and Q22 performs the subtraction process outputs 'V_{th}' at the drain of Q22, as shown in Fig. 9.

In order to validate the operation-ability, functionality, and accuracy of the designed hardware Trojan, an experiment consisting of all elements of HT infection scheme (***RO based heating elements, threshold voltage meter, and the trojan circuit***) is performed. It ascertains whether a triggering signal, ***a shift (increment) in pre-defined threshold voltage***, can be latched or not. Furthermore, in case of being latched, ascertain whether the payload (***accelerated ageing***) of the hardware Trojan gets activated. A controlled temperature environment is ensured using a thermal chamber with an HT infection scheme-implemented FPGA (*28 nm technology node*) placed inside it. The external temperature (*i.e., thermal chamber temperature*) is maintained between 5-10°C (***a typical warship computer control room temperature***). The JTAG interface is used for programming and bidirectional communication between the FPGA and the workstation. Digital oscilloscope, Vivado power analyser, FPGA system monitor, and integrated logic analyser (ILA) are employed to capture the threshold voltage, drain current, and thermal points.

The first stage is the initialization of FPGA under test. This involves the stabilization of the thermal chamber at 5°C, powering up of the target FPGA, and providing an operating voltage of 1.0V. Once powered up, the LUT based ring oscillators implemented to produce heat are enabled. This leads to the second stage where the heat (rise in temperature and a corresponding shift in threshold voltage) generated by the heating elements, spread across the device at locations shown in Fig. 8(b) is continuously measured and logged using the local as well as the system monitor. The temporal change in temperature observed is shown in Fig. 10. As the temperature traverses the primary thermal point of 'T_{p1}' (***60°C***), the changes in threshold voltage 'V_{th}' and 'I_{dd}' are extracted and measured by Threshold Voltage meter. Similarly, the changes are continually observed, and measurements are taken at secondary and tertiary thermal points (T_{p2}-***90°C*** and T_{p3}-***125°C*** respectively). We took 10K samples for each thermal point at all the eight different locations within FPGA. A complete mesh of plot showing the shifts in threshold voltage with change in temperature is given in Fig. 10. In the third stage, these readings are critically analysed for false positives and accuracy for temperature variation and corresponding shifts in threshold voltage as well as 'I_{dd}' to observe the presence of any process variations. Accordingly, three additional runs are undertaken to take further readings and observe intra-run deviations to establish measurement accuracy. During all these three stages, the hardware Trojan trigger circuit remains silent connected with the NAND gate and TSPC PMOS transistors till the time the hardware Trojan trigger circuit experiences a shift in threshold voltage from ***0.45V to 0.63V*** (***40%***) in NAND2 and ***0.67V*** (***50%***) in TSPC

TABLE 2. Hardware trojan triggering analysis in NAND2 logic.

NAND2					
Temp. (°C)	V_{th} (V)	% Shift in V_{th}	I_{dd} (µA)	% Shift in I_{dd}	HT Triggered
5	0.45	0	25	0	Not
10	0.45	0	25	0	Not
60	0.49 (V_{th_ptt})	10	23	8	Not
90	0.63	40	16.25	35	Yes
125	0.76	70	10	60	Yes
150	0.85	90	6.25	75	Yes

TABLE 3. Hardware Trojan Triggering analysis in true single phase clock (TSPC) logic.

True Single Phase Clock (TSPC) Logic					
Temp. (°C)	V_{th} (V)	% Shift in V_{th}	I_{dd} (µA)	% Shift in I_{dd}	HT Triggered
5	0.45	0	25	0	Not
10	0.45	0	25	0	Not
60	0.54 (V_{th_ptt})	20	22	10	Not
90	0.68	50	15	40	Yes
125	0.81	80	8.75	65	Yes
150	0.90	100	3.75	85	Yes

logic. Consequently, the trigger circuit of hardware Trojan causes corresponding significant I_{dd} degradation, as can be seen in Fig. 11(a) and Fig. 11(b) respectively. This, eventually cuts off the V_{DD} connection of the PMOS transistors, which constitute the NAND gate and TSPC. As a result, the whole logic is deactivated, thereby crippling its critical function. The quantitative representation of the percentage shift in threshold voltage (*an increase in this case*) of MOSFETs that triggers the stealthy hardware Trojan is given in Tables 2 and 3.

Before approaching a trigger percentage shift in V_{th}, a gradual increase in signal delays is also observable, for instance, with a ***50%*** shift in the threshold voltage and corresponding ***40%*** shift in I_{dd}, the increase in the rise and fall times from 20.5 ps and 26.7 ps respectively to 22.9 ps and 28.0 ps is recorded. TSPC and NAND circuits remain stable with no triggering of hardware Trojan. However, the slowing down of switching control is observable. As the threshold voltage shift hits ***50%*** of the nominal threshold value of ***0.45V***, the hardware Trojan gets activated. The same is observed for ***70%*** to ***100%*** shifts in the nominal threshold voltage. This experimental result is in consonance with the Monte Carlo simulation carried out by sweeping parameter values using Gaussian distribution. For the simulation purposes, the mean value is set to the nominal threshold voltage value (***0.45V***), whereas the standard deviation ($\pm\sigma$) is kept at $\pm$**0.1V** of the mean value.

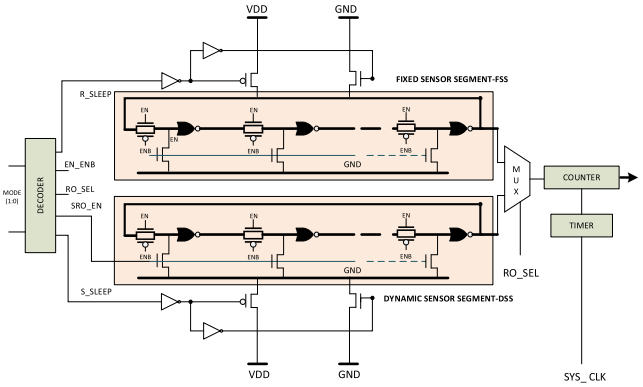


FIGURE 12. The architecture of threshold voltage-aware sensor.

IV. DESIGN AND IMPLEMENTATION OF A THRESHOLD VOLTAGE-AWARE SENSOR

The requirement of a lightweight and highly sensitive sensor for the detection of shifts in threshold voltage much earlier than the triggering of hardware Trojan is a critical design consideration. This is to ensure that the hardware Trojan never gets triggered, provided its presence in FPGA has been accurately assessed. We draw the attention of readers to the not get compromised due to faltering EW-ISPS system dependent on system processor, housing an FPGA. Therefore, the design and implementation of a highly sensitive sensor that detects minor shifts in threshold voltage due to the NBTI effect captures the corresponding frequency shifts and signal path delays and monitors the resultant ageing of the device to provide high confidence in ISPS system performance is paramount. This forms the whole concept of the HT-detection scheme, which is designed and implemented at the recommended placements of security assurance and defence teams, **D** (Fig. 4).

A. THRESHOLD VOLTAGE BASED SENSOR ARCHITECTURE

In continuation to the next stage of the threat model and keeping in perspective the techniques mentioned in [46] and [47], we propose a lightweight sensor that consists of two segments of ring oscillators (ROs), namely the ***'Fixed Sensor Segment (FSS)'*** and the ***'Dynamic Sensor Segment (DSS)'*** as shown in Fig. 12. The fixed sensor segment is designed to experience shifts in threshold voltage at a slower rate as compared to the dynamic sensor segment, which is made to undergo thermal stresses put through the hardware Trojan infection scheme. This must lower the oscillation frequency of the dynamic sensor segment while the fixed sensor segment exhibits a negligible change in its oscillation frequency. With the increasing disparity between the oscillation frequencies of these two segments, the signs of FPGA ageing and hence signal path delays provide a precursor to the inserted hardware Trojan triggering and payload activity.

It is pertinent to mention that the accuracy of a sensor is susceptible to large process variations (PVs) that exist in lower technology nodes. When process variations outpace shifts in oscillation frequency and threshold voltages,

TABLE 4. Binary modes of operation.

Binary Mode	Signals					Explanation
	R_SLEEP	EN	RO_SEL	SRO_EN	S_SLEEP	
00	0	X	X	X	0	RO segments are in dormant phase as their connection to the power and ground line is cut off.
01	0	0	X	1	1	Fixed sensor segment (FSS) remains dormant whereas the dynamic sensor segment (DSS) assumes the threshold voltage-aware mode
10	1	1	0	0	1	Detection and measurement mode activated. Oscillation frequencies/cycle counts of both RO segments are measured.
11	0	1	1	0	1	Detection and measurement mode activated. Oscillation frequencies/cycle counts of both RO segments are measured.

it becomes challenging to differentiate the impact of NBTI from that of the global and local process variations (and this impacts the accuracy of detection and parametric measurements). We overcome this by placing the two segments of ROs very close to each other to zeroise PV and any environmental variation other than the one generated by the hardware Trojan insertion scheme (i.e., the rise in temperature).

The detailed architecture of the proposed sensor is shown in Fig. 12. As can be seen, the dynamic sensor segment is sensitized by introducing a pass transistor between inverters and pulling down the inputs of all inverters to the ground through a network of nMOS transistors. In order to keep all the electrical parameters like node capacitance, resistance, etc. closely matched to the dynamic sensor segment, the same structure is maintained within the fixed sensor segment. Such an arrangement helps ensure that at the time $'t_0'$, when there is no shift in threshold voltage, the difference of oscillation frequency between the two segments is minimal. The only impact observable could be the small variations present between the ROs of the two segments.

In order to implement a specific mode of operation, a decoder circuit is inserted before the two sensor segments to generate the corresponding internal signals, as shown in Table 4. For instance, when enable EN is set to '0', the RO segments start oscillating while the pass transistors stay 'ON.' A timer-controlled counter is placed at the segments' output to enable an instant measurement of their respective cycle counts. For our design of the sensor, four distinct modes of operation, as explained in Table 4, are considered. At mode 1 (**00**), both the segments are inactive or in the dormant phase as their connection to the power and ground line is cut off. This mode is valid for the duration, the heating elements are silent, i.e., during the stabilization phase of the thermal chamber. As the heating element is enabled, and it approaches the primary thermal point ($\boldsymbol{T_{p1}}$**−60°C**), operation mode 2 (**01**) is enforced. In this mode, the fixed sensor segment (FSS) remains dormant (0), whereas the dynamic sensor segment (DSS) assumes the threshold voltage-aware mode (1). Every inverter in DSS is now subjected to dc stress (induced by gradual shifts in threshold voltage) by pulling its input to the ground. This causes changes in its oscillation frequency/cycle count and induces signal delays. When the secondary

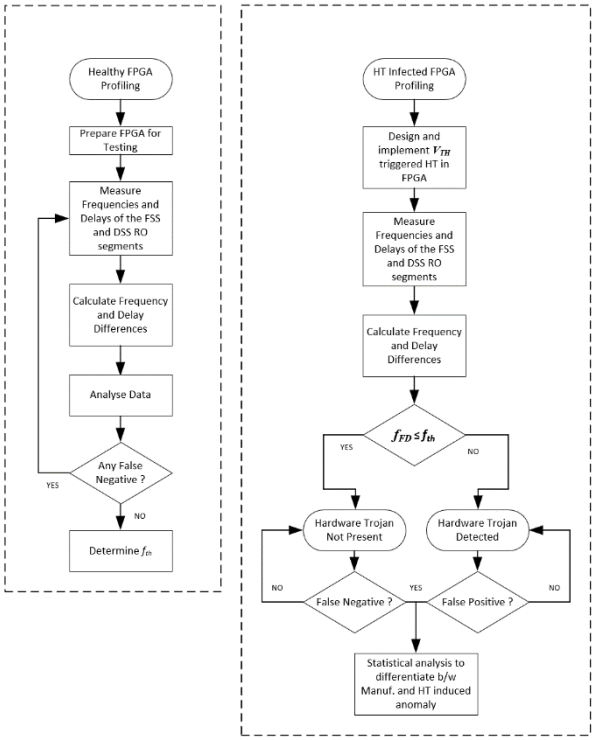


FIGURE 13. Process flows for the identification, authentication, and assessment of Trojan-free and Trojan-infected FPGAs using frequency and delay mapping method.

thermal point $\boldsymbol{T_{p2}}$**−90°C** is reached, the operation modes 3 (**10**) and 4 (**11**) are activated, and oscillation frequencies/cycle counts of both RO segments are measured. This process of measurement continues until the FPGA junction temperature reaches the tertiary thermal point $\boldsymbol{T_{p3}}$**−125°C**. It must be noted here that these measurements are aimed at (1) testing and validating the threshold voltage-aware sensor's efficiency in terms of power and area consumption, (2) determining the frequency threshold of a hardware Trojan-free FPGA at varying locations, and (3) the impact of process variations (PVs) on sensor's accuracy.

B. DETERMINING THRESHOLD FREQUENCY FOR CORRELATION AND AUTHENTICATION

In order to develop a trustworthy threshold voltage triggered hardware Trojan detection scheme, we have defined Trojan-free and Trojan-infected process flows to establish the presence of hardware Trojan in an FPGA. Figure 13 shows the two processes. The main purpose behind the Trojan-free frequency mapping is to determine the threshold frequency $\boldsymbol{'f_{th}'}$ corresponding to pre-Trojan trigger threshold voltage $\boldsymbol{'V_{th\_ptt}'}$ and provide a reference to compare the frequency differences of FSS and DSS $\boldsymbol{'f_{FD}'}$ with it. If $\boldsymbol{'f_{FD}'}$ is greater than $\boldsymbol{'f_{th},'}$ we consider this as an indication of $\boldsymbol{'HT_{Vth}'}$***(threshold voltage-triggered hardware Trojan)*** presence and a precursor to its triggering and payload effect. During the Trojan-free frequency mapping phase, a 28 nm FPGA is used to generate the requisite distributions to determine the

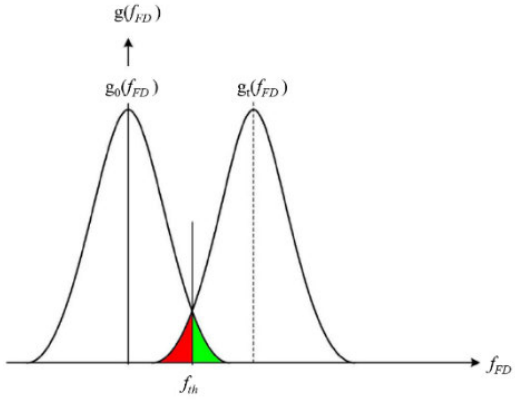


FIGURE 14. Probability density function f_{FD} at times 0 $g_0(f_{FD})$ and t $g_t(f_{FD})$.

threshold frequency *'f_{th}.'* The Trojan-free phase implies that the Trojan circuit is already inserted and present in the FPGA but lying in a dormant state.

Although the two RO segments are placed very close to each other to zeroise the difference of oscillations *'f_{FD}'* between them, yet due to process variations, it will not be zero. Also, a Gaussian distribution of *'f_{FD}'* is observed during the tests. A simplified representation of the two distributions as probability density functions of *'f_{FD}'* at times *'0'* $g_0(f_{FD})$ and *'t'* $g_t(f_{FD})$ is shown in Fig. 14. The frequency differences between the two RO segments *'f_{FD}'* are represented by the x-axis, whereas the y-axis represents the relative distribution function. The overlapping area gives the false prediction of the presence of hardware Trojan or vice versa. The red area ' θ_a' represents the probability of detecting Trojan-infected FPGA as *'HT-free,'* whereas the green area θ_b denotes the probability of identifying the Trojan-free FPGA as *'HT-infected.'* Mathematically,

$$\theta_a = \int_{-\infty}^{f_{th}} g_t\,(f_{FD})\,df_{th} \tag{1}$$

$$\theta_b = \int_{f_{th}}^{\infty} g_0\,(f_{FD})\,df_{th} \tag{2}$$

where $\mathbf{g_0(f_{FD})}$ and $\mathbf{g_t(f_{FD})}$ correspond to the distribution of frequency differences for Trojan-free (dormant) and Trojan-infected FPGAs, respectively. The threshold frequency *'f_{th}'* is considered to be a point where both distributions intersect one another, hence representing the frequency difference that reduces the total probability of error ($\boldsymbol{\theta_a+\theta_b}$).

C. REDUCING THE RATE OF FALSE PREDICTION

When the application risk is as critical as in our ISPS case, it is not prudent to let the false prediction, as identified earlier, result in the system failure by failing the proposed sensor to detect hardware Trojan. The repercussions of such a failure may include the collapse of a defence system of the warship and fatal impact on human and material assets. We have, therefore, devised a process of minimizing (*zeroising*) the level of false prediction of the presence of hardware Trojan and vice versa, as shown in Fig. 15(a)-(c). We observe

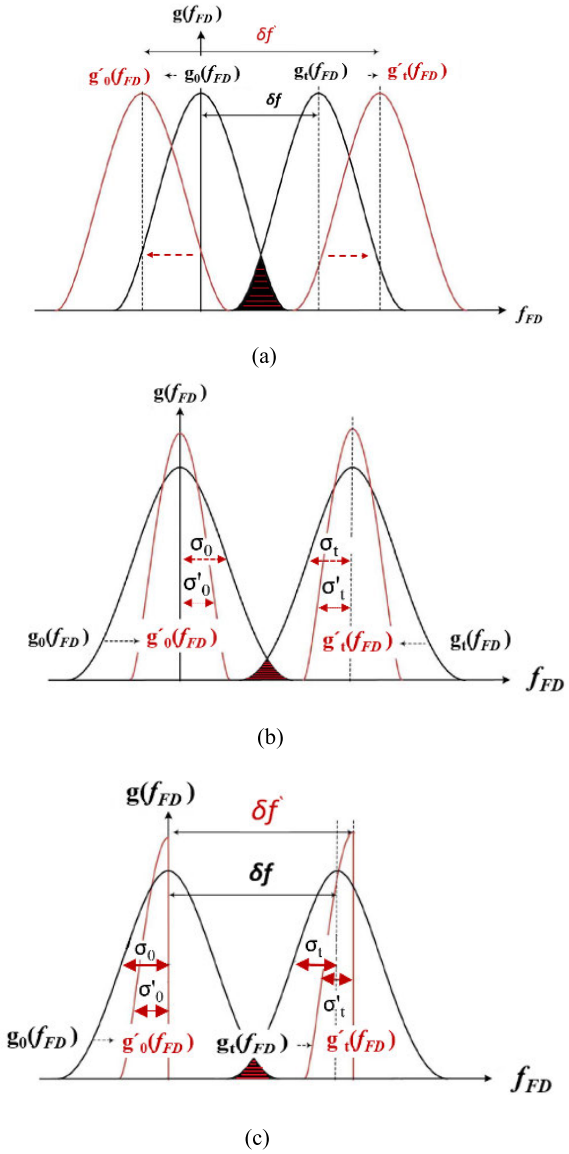


FIGURE 15. Reduction of false prediction - represented by the overlapped area. (a) Moving the FSS and DSS distributions away from their respective positions. (b) Minimizing their spread. (c) Minimal spread with a shift of the mean of FSS and DSS distributions.

that false prediction is generated due to the overlap of FSS and DSS ROs' frequency difference distribution at time *'0'* $g_0(f_{FD})$ and at time *'t'* $g_t(f_{FD})$, which, in this case, is the 'delay' replica of $g_0(f_{FD})$. It implies that if this overlapping region is reduced, the critical issue of false prediction can be resolved.

Accordingly, as a first step, we increase the separation of these distributions, which represents the delay degradation *'δf'*, by shifting the distribution $g_0(f_{FD})$ to the left $g'_0(f_{FD})$ or alternatively shifting the distribution $g_t(f_{FD})$ to the right $g'_t(f_{FD})$ or by implementing both simultaneously as shown in the Fig. 15(a). We observed an improved detection of shifts in frequency corresponding to gradual shifts in the threshold voltage as the distribution $g_t(f_{FD})$ is shifted to the right. Secondly, we consider reducing the spread of FSS and DSS

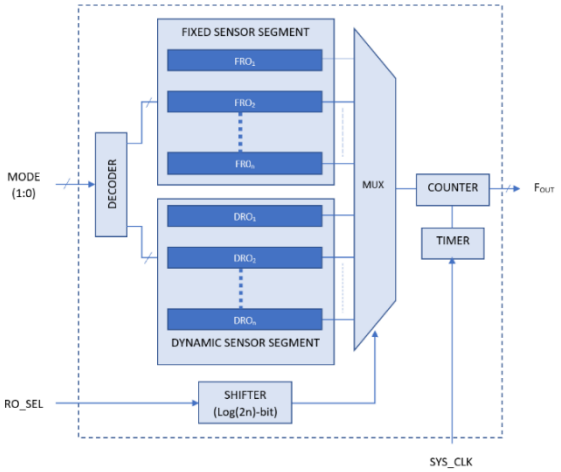


FIGURE 16. Threshold voltage-aware sensor with enhanced detectability of hardware Trojan due to additional RO pairs architecture.

frequency difference distributions. The spread is observed due to the variances of distributions (σ_0^2**and** σ_t^2**).** As can be seen in Fig. 15(b), there is no overlap between $g'_0(f_{FD})$ and $g'_t(f_{FD})$, where $\sigma'_0 < \sigma_0$ and $\sigma'_t < \sigma_t$. This arrangement also helps to minimise the false prediction rate. Thirdly, we reduce the spread and increase the separation of these two distributions simultaneously, as depicted in Fig. 15(c), instead of managing them individually. In such a case, we discard the right-hand side and reduce the spread of $g_0(f_{FD})$ on the left-hand side. It helps reduce the overall spread. The separation, on the other hand, is simultaneously increased by shifting $g_t(f_{FD})$ to the right-hand side. This technique provides the best detection of frequency degradation and hence, the delay - a pointer towards hardware Trojan activity and corresponding ageing of an FPGA under test. For a detailed account of determining maximum frequency degradation through the application of 'Averaging and Selection' methods, please refer to **appendices A** and **B** at the end of this paper.

D. RE-ARCHITECTURING THE SENSOR WITH ADDITIONAL RING OSCILLATOR SEGMENTS

Based on the mathematical mean and variance derivations for FSS and DSS segments with additional RO pairs ***(explained in detail at Appendix A)***, we re-architectured the sensor, as shown in Fig. 16. It consists of the same segments but with two additional threshold voltage shift-aware RO pairs in both. The decision to implement an additional number of RO pairs is primarily aimed at enhancing detectability of abnormal frequency degradation in the shortest amount of time with a negligible false prediction. The results of our experiment show that by the addition of two more RO pairs in both the segments, the detectability of hardware Trojan based on shifts in threshold voltage is unerring.

Looking further at the architecture of the proposed sensor in Fig. 16, it can be seen that the outputs of all the three RO pairs in both the segments are fed to a multiplexer. A shift

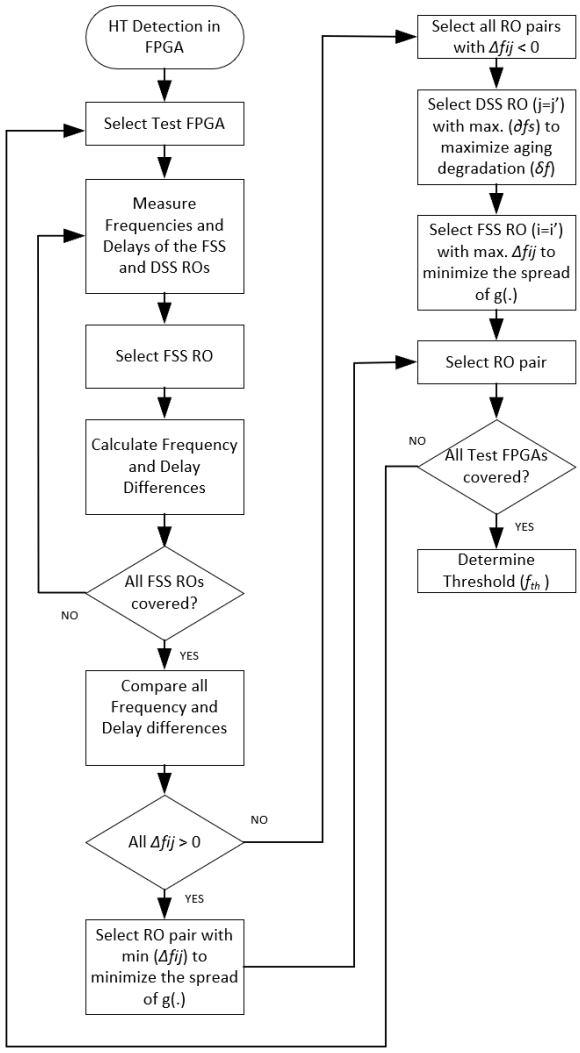


FIGURE 17. Process flow for enhanced detectability of hardware Trojan using optimum-performing RO pairs' selection strategy.

register of $\log_2(2n)$ bit facilitates the Mux. input selection and helps minimise the I/O pin count for the sensor. This register is activated using a 'serial-in RO_SEL' pin. The Decoder, as mentioned earlier, is designed to generate all the internal inputs/signals for the FSS and DSS RO based segments. It is noteworthy that all the RO pairs in each segment utilize the same internal signals generated by the Decoder, and it is not essential to generate the control signals for each RO pair. The operation of the Counter and Timer is the same as elaborated in section IV-A of this paper.

In order to achieve high detection and measurement accuracy, we, besides adopting the averaging strategy, also consider the selection strategy as depicted in the process flow in Fig. 17. The selection strategy implies finding a DSS RO that experiences maximum frequency degradation/delay and hence the ageing due to the NBTI mechanism. For this purpose, the DSS RO pair is compared with the FSS RO pairs even though they remain dormant during normal operations. It is, therefore, essential to find an FSS RO pair that is slower

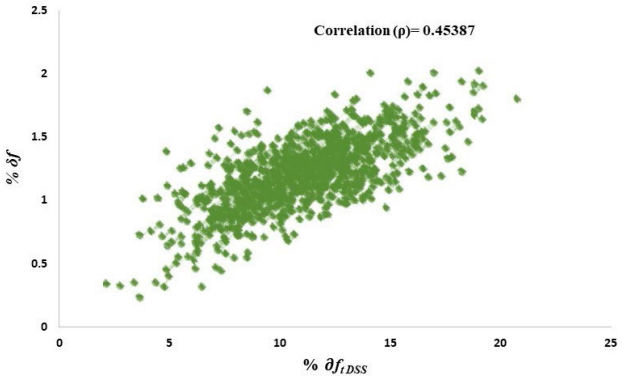


FIGURE 18. Scatter plot of correlation between dynamic frequency degradation (% δf) and percentage frequency difference (% $\partial f_{t\ DSS}$) of DSS ROs (*Refer to Appendix B).*

than the DSS RO pairs during the time ***'0'*** to design a higher sensitivity sensor that enables the detection of hardware Trojan activity well before its onset.

E. SENSOR AND HARDWARE TROJAN DETECTION SCHEME - TESTING AND ANALYSIS

The correct verification of the effectiveness and sensitivity of threshold voltage based sensor for a hardware Trojan detection scheme is, therefore, critical. Consequent to the optimisation of sensor accuracy described in the above section, we implemented the improved sensor design (*with additional RO segments*) in a 28 nm FPGA technology node. The experiment was set up to provide and emulate the ISPS system environment onboard a naval vessel for realistic side-channel measurements. A nominal supply voltage of 1.0V is provided from a benchtop power supply having basic voltage setting accuracy and voltage readback accuracy of 0.03%. With the enabling of heating elements (*following the same phase −1 process with Negative bias '−1.2V' and Tp '60°C'*, as *described in Section IV-A)*, the first set of readings (*including threshold voltage, oscillation frequency/count, and corresponding signal delays*) is taken at stabilised negative bias and primary thermal point, using DL850E ScopeCorder with sample rates up to 100 MS/s.

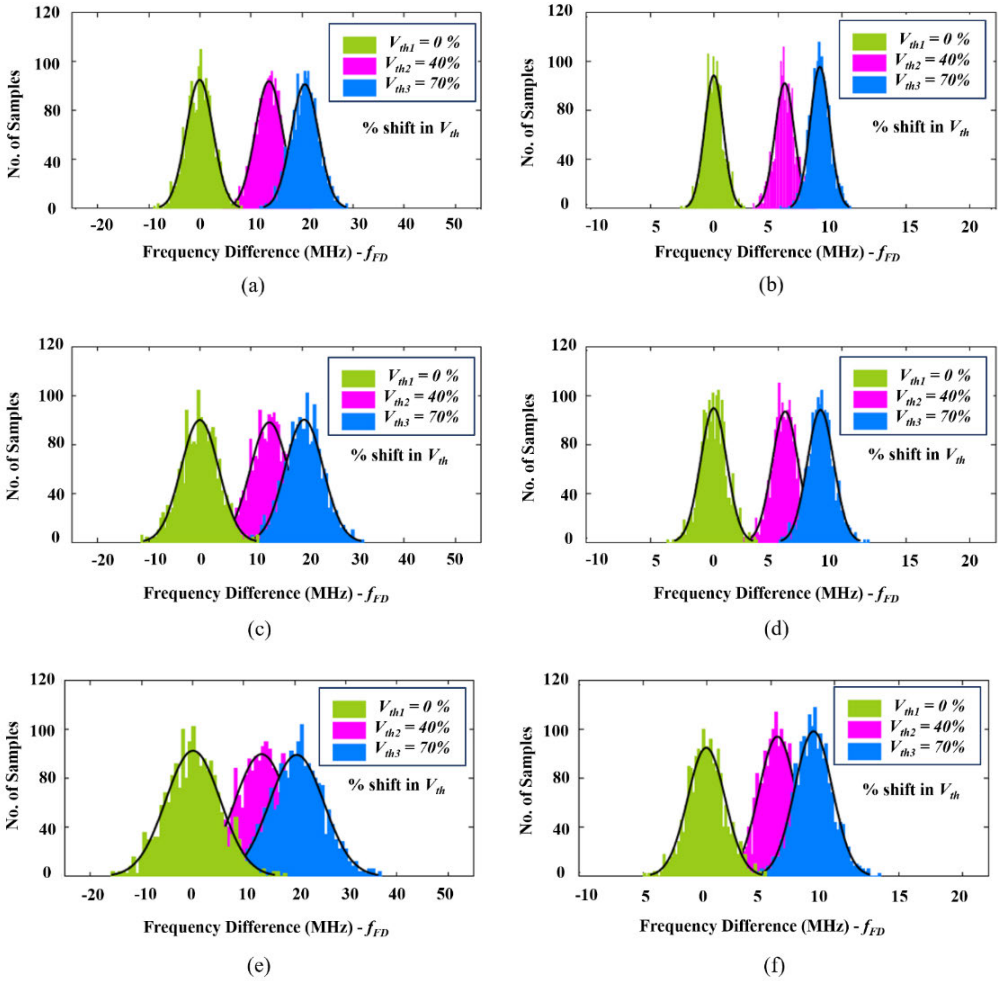


FIGURE 19. Distribution of frequency differences between FSS and DSS, f_{FD}, with percentage shifts in threshold voltage in the presence of process variationsPV_a, PV_b, and$PV_c$and changing number of RO stages (9 and 31) in sensor segments. (a)PV_a: 9-stage RO, (b)PV_a: 31-stage RO, (c)PV_b: 9-stage RO, (d)PV_b: 31-stage RO, (e)PV_c: 9-stage RO, (f)PV_c: 31-stage RO.

TABLE 5. Intra-die process variations - Transistor length and oxide thickness.

Intra-die Process Variations	Parameter	
	Transistor Length (L) %	Oxide Thickness (T_{ox}) %
PV_a	1.5	0.75
PV_b	2.5	1.5
PV_c	8	3.75

Similarly, the experiments were conducted for the second and third phases of the scheme. Although the impact of PVs is minimal as the two sensor segments are placed very close to each other, we did, however, consider the impact of process variations on the detection sensitivity of the sensor in terms of percentage, as given in Table 5.

These tests were repeated to establish the consistency of results and assure the robustness of the developed scheme. The synopsis of test results is given in Fig. 18 and Fig. 19 (a) – (f). The frequency difference of FSS and DSS 'f_{FD}' is represented along the x-axis, and the y-axis represents the frequency of occurrence/the number of test samples. Three different threshold voltage shift states 'V_{th1}, V_{th2}, and V_{th3}' corresponding to 'f_{FD}' are representative of V_{th} distribution.

The green (V_{th1} =0%) distribution plot for 'f_{FD}' is centred at 0 Hz. Whereas, the distributions in pink and blue corresponding to V_{th2} = 40% and V_{th3} = 70% respectively shift to the right. This is because the oscillation frequency/count of DSS slows down and results in a much larger change in frequency difference f_{FD}. With no distinct overlap of distributions (at $V_{th1} = 0\%$ and $V_{th2} = 40\%$ and $V_{th3} = 70\%$), there is a strong indication of the presence of hardware Trojan. We can, therefore, positively detect the presence of hardware Trojans with $V_{th2} = 40\%$ in an FPGA under test (*28 nm node*).

In order to correctly estimate the percentage of false prediction, which is represented by the distributions' overlap, we use Gaussian fit to determine the mean and variance of these distributions to calculate the overlapped area. At this stage, the process variations mentioned in Table 5 are taken into account. These variations being part and parcel of every silicon die, tend to affect electrical parameters invariably from die to die and intra-die as well. With PV_a, we consider the probability of false prediction as negligible, and the same was observed during the test. The measured false prediction rates of the sensor relating to **HT-free** (θ_a) and **HT-infected** (θ_b) FPGA are elaborated in Table 6. These correspond to the process variations mentioned in Table 5. It can be seen that the false prediction rate with PV_c is higher due to a significant difference in frequencies of the 28-nm FPGA under test with a higher percentage of process variations. As a result, the overlapped area between the two distributions grows significantly, thereby reflecting the increase in the probability of error (θ). We provided remediation by placing the two sensor segments very close to each other, as mentioned earlier. Besides, we

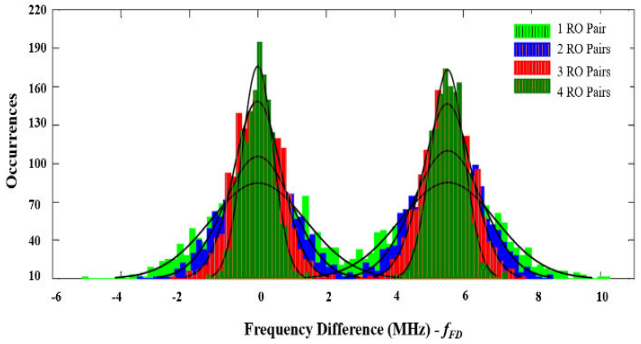


FIGURE 20. Gaussian distribution of frequency difference 'f_{FD}' at PV_c of V_{th}-aware sensor with different number of RO-pairs.

increased the number of RO stages in both the segments from 9 to 31 and then observed any reduction in false prediction rate. A significantly lower false prediction rate is noted (at worst case PV_c– **1.42% to 0.11%**) in the case of θ_b, and a similar trend is noted for θ_a(at worst case PV_c– **1.37% to 0.13%**).

The histogram plot giving the average frequency difference between the FSS and DSS sensor segments for the different number of pairs is shown in Fig. 20. We observe a substantial reduction in the spread of the distributions with the increase in the number of RO-pairs. The separation between the two distributions, however, remains the same. At this point, the threshold frequency f_{th} is measured for all the RO-pairs of the two segments and is found to be equal to 2.5 MHz. It becomes crucial at this stage to analyse the changes in the mean (μ) and variance (σ) values of the frequency difference distribution of sensor segments to estimate the false prediction accuracy to assess any requirement to increase the number of RO-pairs for achieving a negligible false prediction rate. We took the measurements of the mean and variance of different distributions with different numbers of RO-pairs using the '*normfit* MATLAB function' to determine the accuracy of our process flows.

The measured values of the mean and variance are given in Table 7. The analysis revealed an error in the expected value when compared with the actual value (*<0.4% for* μ *and <6% for* σ). In light of this analysis, we created another histogram plot, as shown in Fig. 21(a)-(c), based on the frequency difference between the selected RO-pairs of FSS and DSS sensor segments to determine the most efficient and error-free hardware Trojan detection pair. We observe a significant overlap gap between the two distributions at time $t = 0$ and time t.

Also, the increase in the separation between the distributions is found to be positively correlated to an increase in the number of RO-pairs. The threshold frequency f_{th}, in this case, is measured to be 2 MHz. We found the two RO-pairs combination to be the most appropriate with zero-false prediction. The detection accuracy of the sensor is presented in Table 8.

TABLE 6. False prediction rates (Probability of Error).

No. of RO stages in Sensor Segments	θ_a (%) – Probability of HT-infected FPGA						θ_b (%) – Probability of HT-free FPGA					
	V_{th2}			V_{th3}			V_{th2}			V_{th3}		
	PV_a	PV_b	PV_c	PV_a	PV_b	PV_c	PV_a	PV_b	PV_c	PV_a	PV_b	PV_c
9-stage RO	0.45	2.35	6.29	0	0.12	1.42	0.31	2.19	6.74	0	0.15	1.37
31-stage RO	0	0.22	1.56	0	0	0.11	0	0.25	1.25	0	0	0.13

TABLE 7. Mean and variance frequency distribution of threshold voltage aware sensor.

No. of RO Pairs	g_0 (.)				$g_t = 10^5$ s(.)			
	μ		σ		μ		σ	
	Est.	Meas.	Est.	Meas.	Est.	Meas.	Est.	Meas.
2	0.000	0.012	0.723	0.785	3.213	3.220	0.793	0.887
4	0.000	-0.021	0.524	0.525	3.213	3.220	0.613	0.611
6	0.000	0.001	0.419	0.420	3.213	3.201	0.522	0.538
10	0.000	-0.008	0.400	0.401	3.213	3.242	0.401	0.402

TABLE 8. Analysis of false prediction - Improving sensor accuracy with RO-pairs scaling and selection process.

Percent Shift in Threshold Voltage (V_{th})	RO Pairs (n)	Probability of Error θ_a (%)	Probability of Error θ_b (%)	f_{th} (MHz)
5%	1	13.07	12.85	0.4
	2	5.46	5.51	0.1
	3	2.73	2.77	0
10%	1	9.0	8.9	0.7
	2	2.8	2.79	0.2
	3	1.1	1.15	0
40%	1	5.58	5.38	1.0
	2	1.24	1.12	0.5
	3	0.32	0.34	0
70%	1	3.91	3.75	1.0
	2	0.64	0.58	0.5
	3	0.12	0.14	0
100%	1	1.82	1.65	1.4
	2	0.14	0.14	1.0
	3	0	0	0

The rate of false prediction is calculated as:

$$\theta_a = \frac{no.\ of\ test\ samples\ with\ f_{FD} < f_{th}}{Total\ test\ samples} \times 100\% \quad (3)$$

$$\theta_b = \frac{no.\ of\ test\ samples\ with\ f_{FD} > f_{th}}{Total\ test\ samples} \times 100\% \quad (4)$$

At different threshold voltage shift states, the impact on sensor accuracy with varying number of RO-pairs corresponding to each sensor segment is shown. As mentioned in previous paragraphs, we have implemented a maximum of 3 pairs of ROs each in two sensor segments, FSS and DSS. With a configuration of 2 RO-pairs, we can characterise the sensor to determine threshold frequency ***'f_{th}'*** corresponding to pre-Trojan trigger threshold voltage ***'V_{th_ptt}'*** and provide a benchmark to compare the frequency differences of FSS and DSS ***'f_{FD}'*** with it for the detection of hardware Trojan, once triggered without any probability of error. It is essential to set the threshold frequency cautiously to ensure that the value of the probability of error of FPGAs falsely identified as **HT-free** (θ_a) is similar to the value of the probability of error of FPGAs falsely identified as **HT-infected** (θ_b).

TABLE 9. Area overhead analysis of threshold voltage-aware sensor (S_{Vth}).

Benchmark	Size (No. of Gates)	Sensor [46]	Area Overhead (%) Proposed V_{th}-aware Sensor (31-stages)		
			n=2	n=4	n=6
Vga_lcd	124031	5.23	0.189	0.363	0.93
Ethernet	46771	0.13	0.465	0.996	1.24
DSP	32436	0.19	0.604	1.25	2.32
b15	12562	0.50	1.867	3.387	5.11
b14	8679	0.73	1.258	5.047	2.89
spi	3277	1.92	1.474	3.37	3.90
i2c	1142	5.52	1.23	3.35	5.12

F. AREA OVERHEAD ANALYSIS

The implementation of a threshold voltage triggered hardware Trojan detection scheme is optimized to utilize minimum resources of 28 nm technology node FPGA. Accordingly, the area overhead analysis of both the infection and detection schemes is shown in Table 9. We implemented IWLS 2005 benchmarks of various sizes from low to high to assess the area overhead - the ratio of the size versus area of the sensor with the size versus area of the benchmark. As is evident, when used with a 31-stage sensor in HT detection scheme, the area overhead is approximately $\mathbf{1.25\%}$ ***for*** $\boldsymbol{n = 2}(2\ RO - pairs)$ for smaller sized benchmarks like i2c, spi, and b14. We observe that it does not impact the overall area of small as well as medium and larger designs, implemented for heavy systems like the system processor module of ISPS, in our case. On average, the overall area occupied by the HT-detection scheme is measured to be $\mathbf{125}\mu\mathbf{m^2}$, whereas the power consumption reads $\mathbf{3.8}\mu\mathbf{W}$,

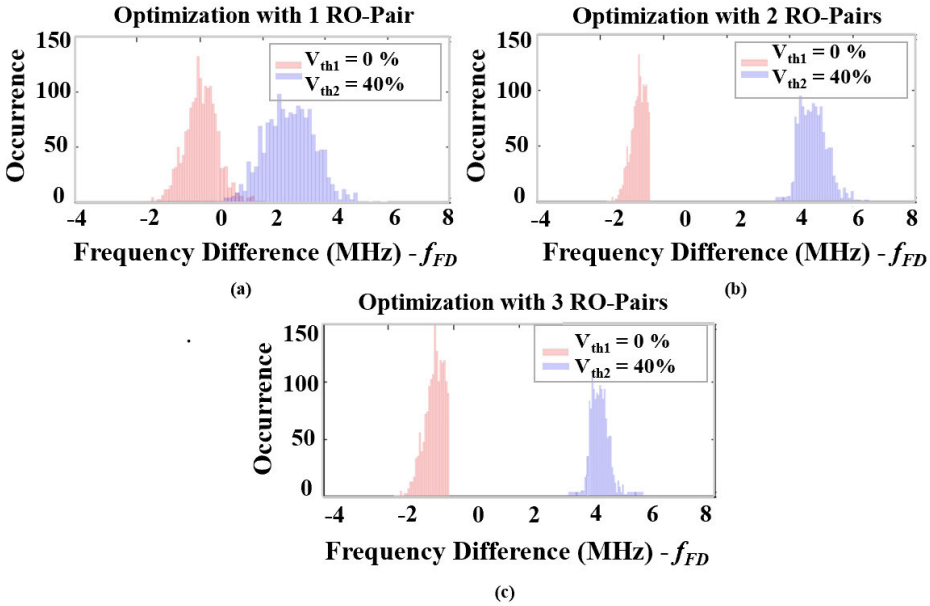


FIGURE 21. Histograms of frequency difference distribution f_{FD} at PV_c of V_{th}-aware sensor with different number of RO-pairs. (a) Optimization with 1RO-pair. (b) Optimization with 2 RO-pairs. (c) Optimization with 3 RO-pairs.

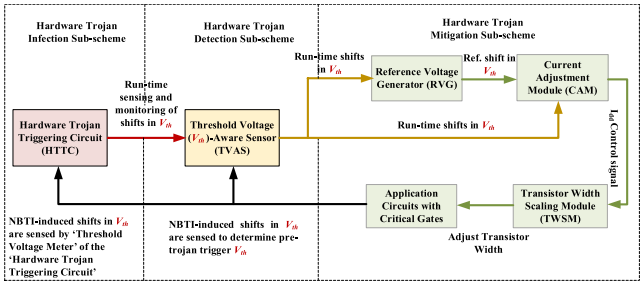


FIGURE 22. Block diagram representation of FPGA security scheme highlighting hardware Trojan mitigation sub-scheme.

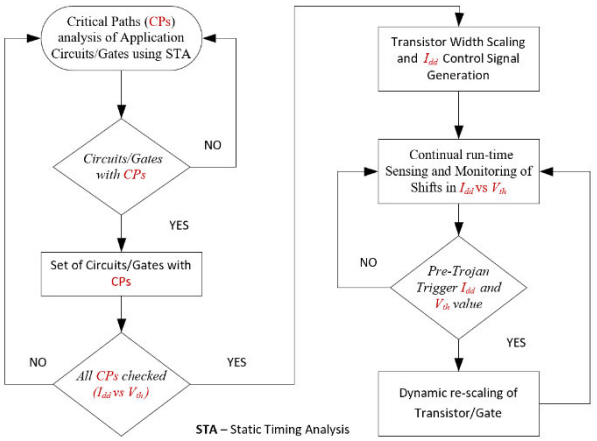


FIGURE 23. The process flow of hardware Trojan mitigation scheme.

which is considered compatible with the designs discussed in Section-II.

V. MITIGATING THE IMPACT OF THRESHOLD VOLTAGE-TRIGGERED HARDWARE TROJAN

The final proposition of FPGA security scheme (Fig.3) is the design and implementation of hardware Trojan mitigation strategy. We propose a circuit design technique, which endures threshold voltage-triggered hardware Trojans. The internal module structure and control process flow devised. for this purpose are depicted in Fig. 22 and Fig. 23 respectively. For this scheme, we target the monitoring of drain current ***'I_{dd}'*** as a parameter that contributes to performance degradation as a result of shifts in threshold voltage. A mechanism is proposed whereby a change in the threshold voltage is sensed and a corresponding adjustment in I_{dd} is made to compensate for current variations in critical circuit nodes implemented in FPGA.

The main elements added to form the mitigation scheme are the 'Current Adjustment Module,' 'Reference Voltage Generator,' and the 'Transistor Width Scaling Module.' IWLS 2005 benchmark 'vga_lcd' is used as a test circuit implemented in 28-nm FPGA to validate the HT mitigation scheme. It also includes the process of pinpointing the potential critical gates that experience frequency degradation due to the impact of NBTI through shifts in threshold voltage. The Current Adjustment Module (CAM) gauges the acceptable limits and ranges of shifts in threshold voltage, fanned out by the sensor (in our case, the V_{th_ptt}). If V_{th_ptt} *(pre-trojan trigger threshold voltage)* is out of the acceptable limit, the control signal is given to the Transistor Width Scaling Module (TWSM), which increases the transistor width to counter the

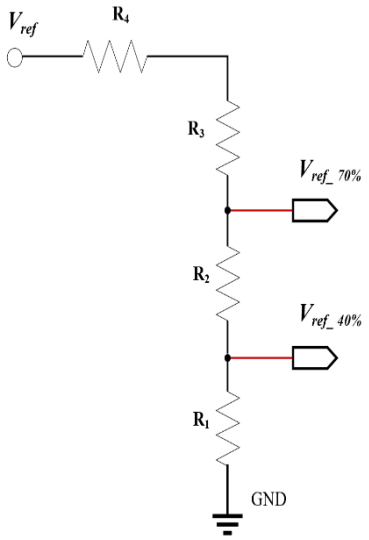


FIGURE 24. Resistive voltage divider for reference voltage generator (RVG).

excess threshold voltage shift and prevent the triggering of hardware Trojan.

A. EARMARKING THE POTENTIAL CRITICAL GATES

We implemented the IWLS 2005 benchmark 'vga_lcd' in 28-nm FPGA using the Vivado design suite and applied the algorithm defined in [48] to pinpoint its potential critical gates using static timing analysis. We conclude that only 2.5% of the total gates are identifiable as the potential critical gates, based on the worst-case frequency/delay degradation. The worst-case degradation is set against the V_{th_ptt}. Accordingly, a reserve transistor width is allocated to the earmarked critical gates to increase $'I_{dd}'$ and counter the impact of the increased threshold voltage. The details of the implementation are described later in section V-D.

B. REFERENCE VOLTAGE GENERATOR

The measurement of the threshold voltage is done using 'Threshold Voltage Meter'(Fig. 9). Although we have used the percentage frequency differences corresponding to specific threshold voltage shifts in the HT detection scheme, we consider it prudent to quantify the impact of shifts in threshold voltage due to NBTI, while devising HT mitigation scheme. In this regard, we propose the implementation of a 'Reference Voltage Generator' comprising a resistive-based voltage divider. The schematic of the generator is shown in Fig. 24. While calculating the reference voltages, the effect of resistive tolerance is taken into account. Resultantly, for the threshold voltage shifts of 40% and 70%, for instance, we represent them correspondingly as $V_{ref\_40\%}$ and $V_{ref\_70\%}$. In order to determine the effect of resistive tolerance variations, we carried out Monte Carlo simulation, taking into account the process and environmental variations as well. A maximum change in reference voltage ΔV_{ref} of less than 4mV is observed at a **worst-case** resistive variation of $\pm$ 5%. Whereas at **nominal** ($\pm$ **3%**) and **best case** ($\pm$ **0.5%**) variations, ΔV_{ref} **of less than 2mV** and **0.75mV** respectively are noted.

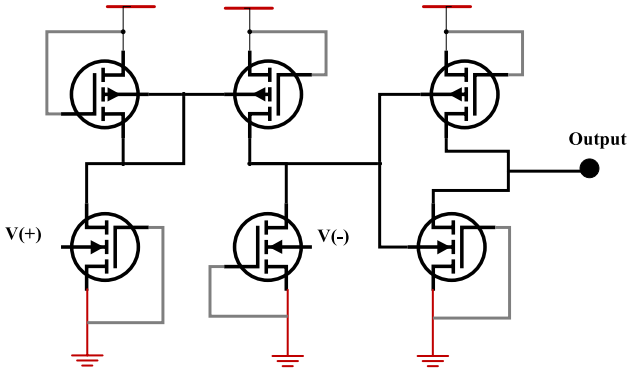


FIGURE 25. A Comparator circuit with current-mirror based differential amplifier.

C. CURRENT ADJUSTMENT MODULE

Since the shift in threshold voltage of a PMOS device results in the reduction of drain current and the subsequent slowing down of the circuit speed, it is possible to reverse or mitigate this phenomenon by increasing the drain current. In order to achieve this, a comparator circuit comprising current-mirror based differential amplifier is implemented as a current adjustment module. The schematic of this module is shown in Fig. 25. Here, the output of the HT detection scheme and the reference voltage generator drive the inputs of the current adjustment module. A control signal from the current adjustment module is provided to the TWSM module, which subsequently increases the width of the transistor to counter the frequency degradation/delay impact of the NBTI mechanism.

In order to check the operation-ability of this module, we induce a fractional change at the inverting and non-inverting inputs of the comparator, as shown in Fig. 26. When the voltage on the inverting terminal of the comparator is made higher as compared to its non-inverting terminal, the comparator switches to logic '0' and vice versa. We considered the impact of process variations as well and found the comparator sensitive up to **1.5mV** of variation between inverting and non-inverting terminals.

D. TRANSISTOR WIDTH SCALING MODULE

Increasing the transistor width to let more current pass through the transistor can be implemented as a countermeasure against the threshold voltage triggered hardware Trojans to mitigate the latency induced by the shift in threshold voltage [49]. However, designing transistor width increment as a one-time design rule makes it ineffective against the long-run online performance degradation caused by NBTI ageing mechanism [49]. Also, device upsizing could inflict constraints on the design specification during the design stage. Many design metrics, like impedance matching and Q point of V-I curve, may be affected, which may result in excess drain current values. It is for these reasons, we propose

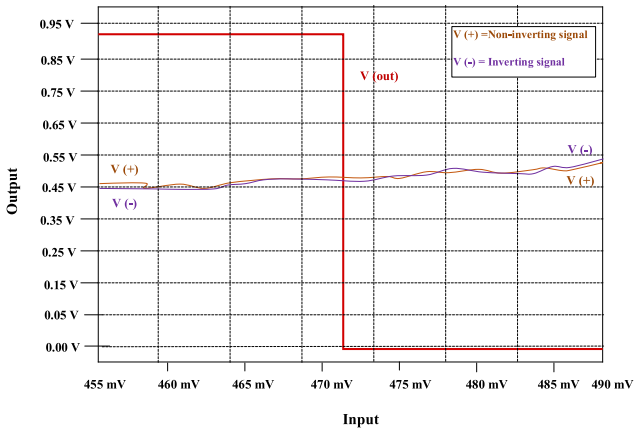


FIGURE 26. Input / Output response of a comparator.

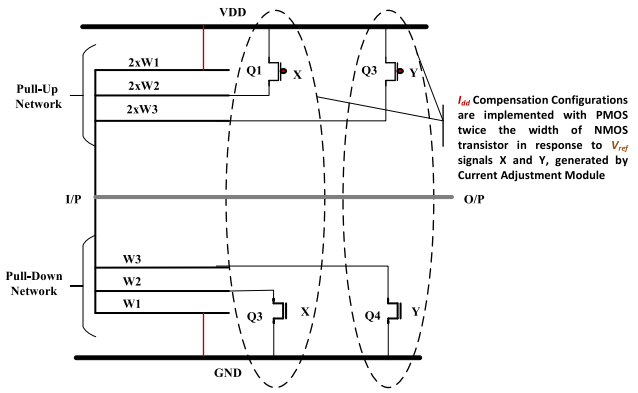


FIGURE 27. Online transistor dynamic scaling using pull-up and pull-down networks.

a hardware Trojan mitigation scheme that adjusts the width of transistors dynamically (i.e., during run-time) and named as **'Online Transistor Dynamic Scaling (OTDS)**. We divide OTDS into two implementation phases as follows:

1) DESIGN PHASE

In the design phase, we define the dimensions of the 2% of identified critical gates of IWLS 2005 benchmark 'vga_lcd' in-line with its I/O functional specification. Additionally, we provide the threshold voltage compensation dimensions/sizing as a backup for the potential critical gates. As per the design, the dimensions of the transistor forming the critical gate remain fixed until it is sensitized by a significant NBTI impact on the design embedded in FPGA.

2) DYNAMIC PHASE

As mentioned in the above paragraphs, when threshold voltage begins to change (increase with NBTI), a runtime decision will be asserted to increase the width of the critical transistors. With an increase in transistor width, the device is supported with a corresponding increase in its drain current and hence, balances and mitigates the impact of threshold voltage shifts.

The concept is illustrated in Fig. 27. It shows an inverter having a PMOS double the size of its NMOS counterpart. Under the normal situation, the pull-up network possesses two unconnected parallel widths (2xW2 and 2xW3). Similarly, the pull-down network consists of two unconnected parallel widths (W2 and W3). We gated the additional PMOS widths, 2xW2 and 2xW3, using transistors Q1 and Q3, respectively. Similarly, the additional NMOS widths W2 and W3 are also gated using the transistors Q2 and Q4 respectively. The transistors Q1 and Q2 are set to share the same triggering signal from node X whereas Q3 and Q4 share the identical signal from node Y. Under the normal condition, defined as $\boldsymbol{V_{th} < V_{th\_ptt}}$, all these transistors remain dormant ('Off State') and are considered to be a unit sized transistors. As the threshold voltage is shifted ($\boldsymbol{V_{th} \geq V_{th\_ptt}}$) with bias and temperature stressed NBTI, the OTDS technique tries to compensate its impact by selecting transistors of larger widths. At this stage, the reference voltage generator provides steps of percentage voltage corresponding to percentage shifts in threshold voltage. When an increase of 30% in the threshold voltage of the PMOS transistor is reached, the transistor width is incremented to counter the shift in threshold voltage to prevent HT triggering.

It is vital to have an accurate reference voltage step generation for effective mitigation of the increased threshold voltage and the frequency/delay degradation of the circuit application. For that purpose, we assume the reference voltages to be fixed and the run-time or dynamic state decision is made using the values of threshold voltage measured by the HT detection scheme sensor. During the experiment, we observe that as the threshold voltage rises by 5%, the current adjustment module with a corresponding reference voltage (V_{ref}) generates a signal X, which activates the transistors Q1 and Q2 and turns them 'ON.' At this point, the width of the Pull-Up network, shown in Fig. 27, increases by 2xW2 and so does the width of the Pull-Down network by W2. In the same way, at some instances of the time interval, the signal Y gets triggered with a specific reference voltage, which in turn, activates the transistors Q3 and Q4, having widths as shown in Fig. 27. This leaves the Pull-Up and Pull-Down networks with improved speed and stability.

VI. IMPLEMENTATION AND OPTIMIZATION OF HARDWARE TROJAN MITIGATION SCHEME

It is well established that the drain current **'$\boldsymbol{I_{dd}}$'** and the response time of a MOSFET are directly proportional to its width. Therefore, increasing the transistor's width will subsequently increase the drain current as well as its response time. So, in order to double the transistor width, we may use an equal width transistor to widen the MOSFET by sharing the drain and source terminals between MOSFETs. It also helps in minimising the layout area.

Before deciding the extent of increasing the width of the transistor to reverse current reduction due to NBTI, we quantify the reduction in drain current **'$\boldsymbol{I_{dd}}$'**. Accordingly, we measure **'$\boldsymbol{I_{dd}}$'** at 0%, 10%, 30%, 60%, and 90% of shift in $\boldsymbol{V_{th}}$. The measurement results are listed in Table 10. Based upon these measurements, a width-based parametric analysis of the

TABLE 10. Measured values - PMOS I_{dd} reduction with increase in V_{th}.

% age Shift in Threshold Voltage V_{th} (Increment)	PMOS Drain Current I_{dd} (µA) (28 nm – V_{gs}=0.4V and V_{ds}=0.9V)
0	25
10	20
30	13
60	7
90	2.5

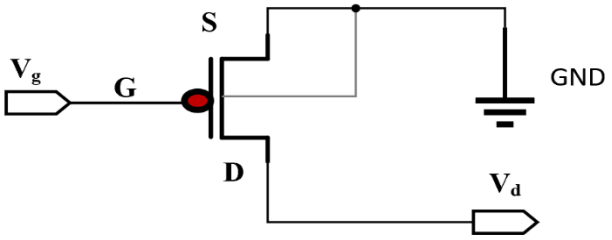


FIGURE 28. Circuitry for transistor width parametric analysis.

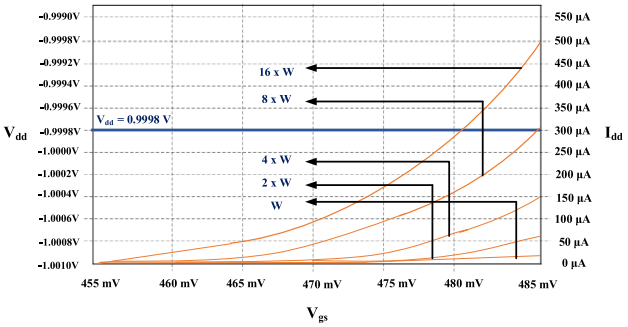


FIGURE 29. I_{dd} vs V_{gs} curves showing online transistor width increment to compensate for threshold voltage-triggered hardware Trojan (HT_{Vth}) attack.

PMOS transistor is undertaken to make a correct assessment of the extent of its width increment required to reverse ***'I_{dd}'*** reduction, corresponding to percentage shifts in V_{th}. This analysis is enabled by the circuitry shown in Fig. 28. As can be seen, we kept the gate and source voltages of the PMOS transistor constant at −1V and 0V, respectively and noted the variation in width (W) of the transistor. The results are shown in Fig. 29. It is evident that for a given gate and source voltages, the drain current increases two-fold as the width of the PMOS device is doubled. So, accordingly, we come up with the requisite percentage of width increment, which is added in parallel for each value of shift in threshold voltage to increase the transistor's width and the current flow through it. The implementation of this scheme is elaborated in Fig. 30.

We employ the unit size transistor as a switch to manage and control the connectivity of a transistor width for compensation. As seen in Fig. 30, Q1 represents the critical gate, and Q2, Q3, and Q4 are the widths reserved to compensate for the reduction of ***'I_{dd}'*** due to percentage V_{th} shifts. As mentioned earlier, the sizes of Q2, Q3, Q4, and Q5 are defined at the design phase. The same are given in Table 11.

In order to validate the mitigation scheme, the circuitry in Fig. 30 is applied to a flip flop with true single-phase

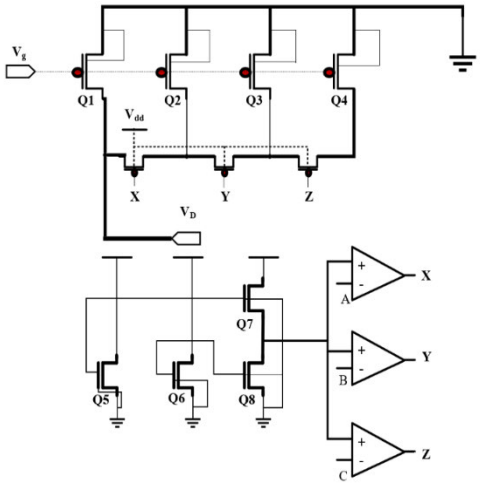


FIGURE 30. Threshold voltage-triggered hardware Trojan mitigation circuitry of the 'HT-Mitigation Sub-scheme.

TABLE 11. Measured values - width increment (Fanout-4) with shifts in V_{th}.

% age Shift in Threshold Voltage V_{th} (Increment)	PMOS Drain Current I_{dd} (µA) (28 nm – V_{gs}=0.4V and V_{ds}=0.9V)	Transistor Width Increment (FO-4)
0	25	NR
10	20	≃1.0
30	13	≃1.2
60	7	≃1.5
90	2.5	≃2.2

TABLE 12. Timing delays in TSPC due to V_{th} -triggered hardware Trojan payload.

V_{th}(V)	% age Shift in Threshold Voltage V_{th}	Rise Time Delay (ps)	Fall Time Delay (ps)
0.450	0	22.5	28.7
0.495	10	23.1	30.5
0.585	30	25.7	32.3
0.720	60	29.4	35.6
0.855	90	34.7	40.5

clocking function. We measure the rise and fall times of the flip flop as they change with changes in the threshold voltage. The results show an increase in the rise and fall times with an increase in V_{th} shifts. The exact values are covered in Table 12. We observe that as a result of this increase, momentary state transitions occur in FSM, which may lead to changing the output state. Also, we note that as the duration of this output state is extended, it gets latched and may result in the activation of malicious and stealthy hardware Trojan. This, however, is prevented by increasing the device width and resultantly, the triggering signal for the Trojan is silenced.

In a nutshell, adding extra reserve width for Pull-Up and Pull-Down network in the design phase provides a viable mitigation technique, which increases the transistor width dynamically during the run-time.

VII. COMPARATIVE ANALYSIS WITH CONTEMPORARY MITIGATION TECHNIQUES

We have presented a holistic FPGA security scheme to detect and mitigate the ingress of threshold voltage triggered

TABLE 13. Area and Power consumption comparison of the proposed Threshold Voltage (V_{th}) -shift based HT Mitigation scheme.

Mitigation Method	Area Utilization (unit sq.)	Area Difference	Power Consumption (μW)	Power Consumption Difference
Omana et al [50]	98	(-) 47 %	16.2	(-) 66 %
Wang et al. [51]	90	(-) 43 %	17.5	(-) 68 %
Bowman et al. [52]	86	(-) 40 %	15.0	(-) 63 %
Vazquez et al. [53]	78	(-) 34 %	15.9	(-) 65 %
Mintarno et al. [54]	75	(-) 31 %	15.8	(-) 65 %
Cao et al. [55]	63	(-) 17 %	15.7	(-) 64 %
Khatib et al. [56]	65	(-) 20 %	17.2	(-) 68 %
Proposed	***52***	-	***5.5***	-

hardware Trojans in its fabric. In doing so, we have designed, implemented, and validated HT-infection, HT-detection, and HT-mitigation schemes, with novel sensing and monitoring elements. We have highlighted its significance in the ship-defence environment by providing a threat scenario/model based on an 'Integrated Self-Protection System (ISPS).' This is a unique effort that puts forth an integrated approach towards visualising and addressing a probable hardware Trojan presence in a security-sensitive and mission-critical defence system with accurate and resource-efficient detection and mitigation circuitry in a 28 nm technology node based FPGA.

As discussed in section II, a significant amount of research work has been undertaken to develop effective methods and circuits. In this section, we make a comparative analysis of our work with other existing methods for the mitigation of the NBTI effect in integrated circuits. For instance, in [50], the adaptive clock scheme entails increasing the clock time to address the worst-case performance (in terms of signal path time delays) degradation due to NBTI. This scheme is, however, hardware-intensive with a high area overhead. Also, it degrades the device performance as a result of time guard banding. Another technique [35] implies the replacement of aged gates to reverse delay degradation but, again, it results in high area overhead. Our work, on the contrary, addresses performance degradation by changing the transistor width dynamically. This entails low area overhead and enhanced device performance.

In another scheme [51], device ageing due to NBTI is countered through standard-cell sensor-facilitated measurement of frequency degradation. It is followed by inducing additional timing margin for the critical path to prevent device failure due to continued ageing. However, the provision of redundancy in terms of extra timing margin is not always valid. Moreover, such kind of schemes is resource-intensive with increased area overheads–an undesired feature in modern technology nodes.

Table 13 summarises the analysis in terms of efficiency with respect to area overhead and power consumption. We find the HT-mitigation component of our FPGA security scheme more resource-efficient with reduced power consumption. It augments the device performance by zeroing the impact of shifts in threshold voltage through responsive and dynamic scaling of transistor width rather than the replacement of the gate/transistor.

VIII. CONCLUSION

The miniaturised form factor of modern FPGAs provides enhanced performance as compared to their predecessors. However, high-temperature stresses coupled with longer heat dissipation paths may cause undesired stochastic variations like signal delays. Primarily, this is attributable to the negative bias temperature instability (NBTI) ageing mechanism that comes into play as a result of elevated temperature and negative bias stress conditions.

Consequently, the threshold voltage increases, which in turn, leads to reduced drain current and delay degradation.

Keeping the aforementioned in perspective, we have investigated the impact of threshold voltage shifts due to the degradation mechanism of NBTI in a 28 nm technology node and constructed an FPGA security scheme around it to counter potential hardware Trojan (HT) threats. The development of a threat scenario/model encompassing a naval warship's integrated self-protection system (ISPS), with its processor module in focus, reinforces the need for a holistic approach to hardware Trojan threats. We have shown how a rogue element in a design house can make use of knowledge about the shifts in threshold voltage of a PMOS transistor to design and implement a stealthy hardware Trojan scheme comprising heating elements, threshold voltage meter, and the Trojan circuit. The area and power consumption for this scheme are kept as low as $\mathbf{50}\mu\mathbf{m}^{\mathbf{2}}$ and **1.05** μ**W** for NAND2 and $\mathbf{75}\mu\mathbf{m}^{\mathbf{2}}$ and **1.25**μ**W** for TSPC, with the hardware Trojans triggering at **40%** and **50%** of the shift in threshold voltages, respectively. It results in the total collapse of the circuit functionality, thereby confirming the paralysing effect it can have on the ISPS system capability of a warship. Acting as a defender, we have created hardware Trojan detection and mitigation schemes as an integral part of the overall FPGA security scheme. The HT-detection scheme is composed of a highly sensitive (**100 KHz/0.5 mV**) ring oscillator pair-based sensor. It measures frequency degradation in a dynamic sensor segment (DSS) RO pair equivalent to the shifts in threshold voltage and compares it with the fixed sensor segment (FSS). The sensor is tested and calibrated to detect frequency degradation at the pre-Trojan Trigger

threshold voltage 'V_{th_ptt}' and Trojan Trigger threshold voltage 'V_{th_tt}.' The detection and measurement accuracy is achieved by reducing the false prediction rate to zero. Area overhead of $\mathbf{125}\mu\mathbf{m^2}$ and compatible power consumption of $\mathbf{3.8}\ \mu\mathbf{W}$ are noted for the HT-detection scheme.

The final part of our FPGA security scheme is HT-mitigation by online transistor dynamic scaling (OTDS). Here, we leverage the reduction in drain current with an increase in threshold voltage to dynamically adjust the transistor width and reverse the HT triggering process. Post parametric analysis of the changes in the transistor width, we conclude that increasing the transistor width improves its drain current flow, which in turn, helps maintain the performance of the FPGA and avoid HT triggering. We correlated and back annotated the requisite increment/decrement in the transistor width to compensate for the drain current loss due to shifts in threshold voltage. Accordingly, a range of transistor widths that compensates for the reduction in drain current has been determined in the FPGA under test. This HT-mitigation scheme occupies an area of $\mathbf{150}\mu\mathbf{m^2}$ with power consumption at $\mathbf{5.5}\mu\mathbf{W}$.

The whole FPGA security scheme is built on changes in the threshold voltage of the PMOS transistor. It provides a unique and integrated strategy for thwarting the probable infection of threshold voltage-triggered hardware Trojans in advanced reprogrammable devices used in security-critical defence systems. In the future, we intend to: 1) study the impact of PBTI in NMOS transistors in conjunction with NBTI and design PBTI based hardware Trojan, 2) extend the scope of FPGA security scheme validation to more complex applications, 3) investigate the health of reprogrammable devices under the influence of such hardware Trojans, and 4) devise AI techniques to provide accurate FPGA prognostics.

APPENDIX A
IMPROVING THE HARDWARE TROJAN DETECTABILITY

Keeping the patterns of false prediction in perspective, we consider improving the proposed sensor's detection sensitivity by adding two additional pairs of ROs to each of the sensor segment (*Fixed and Dynamic*). The frequencies of all these pairs of RO segments are measured consecutively, during different thermal cycles. Subsequently, the average of FSS *(Fixed Sensor Segment)* and DSS *(Dynamic Sensor Segment)* frequencies is calculated to determine the presence of malicious hardware Trojan.

A. SPREAD REDUCTION BY AVERAGING METHOD

Assuming there is n number of ROs in the fixed and dynamic sensor segments, their respective frequencies can then be considered as random variables and denoted by $a_1, a_2, \ldots, a_n$ and $b_1, b_2, \ldots, b_n$, respectively. as the distribution of $g_0(f_{fd})$ depends upon the frequency differences of both the fixed and dynamic sensor segments; we can derive the following equation:

$$X_i = A_i - B_i \tag{5}$$

In this equation, $X_i s$ is Gaussian, as both the $A_i s$ and $B_i s$ are Gaussian. We further assume the variables A and B to have the same mean and variance, as all the RO segments undergo the same process variations. The aim is to determine the mean and variance of a newly formed random variable Z_n. Mathematically, this can be represented as follows:

$$Z_n = \frac{1}{n}\sum_{i=1}^{n} \mathrm{Ai} - \frac{1}{n}\sum_{i=1}^{n} \mathrm{Bi} \tag{6}$$

$$= \frac{1}{n}\sum_{i=1}^{n} (\mathrm{Ai} - \mathrm{Bi}) = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{7}$$

The resultant random variable Z_n will be, therefore, Gaussian as all the $X_i s$ are Gaussian. Based on this, the mean and variance are expressed in the following mathematical form:

$$E[Z_n] = E[\frac{1}{n}\sum_{i=1}^{n} X_i] = \frac{1}{n}\left(\mathrm{E}\left[\sum_{i=1}^{n} X_i\right.\right.$$

$$= \frac{\mathrm{n}x\mu}{n} = \mu \tag{8}$$

$$var(Z_n) = var\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right.$$

$$= var\left(\sum_{i=1}^{n} \frac{X_i}{n}\right)$$

$$= \frac{1}{n^2}\sum_{i=1}^{n} var(X_i) + \frac{1}{n^2}\sum_{i\neq j} cov(X_i, X_j)(9) \tag{9}$$

In the above equations, $E[Z_n]$ is the expected value of the random variable Z_n- equal to the mean of a Gaussian random variable. Whereas $var(Z_n)$ represents the variance of the random variable Z_n and $cov(X_i, X_j)$ is the covariance between the random variables X_i *and* X_j. In this mathematical model, we assume the frequencies of all the RO segments to be independent so that the random variables $X_1, X_2, \ldots . X_n$ also become independent. It, therefore, results in all the covariances in (9) becoming zero.

$$var(Z_n) = \frac{1}{n^2}\sum_{i=1}^{n} var(X_i) = \frac{nx\sigma^2}{n^2} = \frac{\sigma^2}{n} \tag{10}$$

Keeping the above equation (9) in view, the mean (μ) and the standard deviation (σ) of Z_n can be derived as follows:

$$\mu Z_n = \mu \tag{11}$$

$$\sigma Z_n = \frac{\sigma}{\sqrt{n}} \tag{12}$$

As can be seen in (8) and (11) , the mean of the average difference Z_n remains unchanged when compared with each X_i. On the other hand, the variance of Z_n is dependent on $\sqrt{n}$. A similar derivation is carried out to estimate the resultant mean and variance for the distribution at time t, $g_t(f_{FD})$. We, therefore, infer that the overlapping area between the two distributions can be reduced to an almost negligible amount by adding additional RO pairs to both the fixed and dynamic segments, as is evident from Fig. 15(b).

APPENDIX B
DETERMINING MAXIMUM FREQUENCY DEGRADATION

An accurate and precise capturing of frequency degradation in ring oscillators is key to the correct and authentic assessment of hardware Trojan's triggering, its impact, and a reliable measure of the sensor's sensitivity. We, therefore, experimented to determine the maximum frequency degradation experienced by DSS RO pairs when negative bias and elevated temperatures are applied as per the hardware Trojan insertion scheme described in section III 24 and coarse as well as fine stretching operations (stress-time) used in [57] to minimise measurement errors. We observe how the frequency degradation (with subsequent delays and ageing), δf, changes with the percentage frequency differences at varying threshold voltages. A total of 10K samples were taken at each thermal (60, 90, and 125°C) and negative bias (−1.2V, −1.4V, and −2.0V) points. The scatter plot of frequency degradation δf against frequency difference ∂f_{tDSS} at time t is shown in Fig. 18, where $\partial f_{tDSS} = (ft_{(-2.0V)} - ft_{(-1.4V)} - ft_{(-1.2V)}) / ft_{(-1.2V)}$. As is evident, $ft_{(-2.0V)}$, $ft_{(-1.4V)}$, and $ft_{(-1.2V)}$ are the frequencies of DSS RO pairs that are exposed to negative bias and increasing temperature stresses. A positive correlation (ρ) for frequency degradation and normalised frequency differences is observed that indicates the ageing and delay degradation in this specific threshold voltage triggered hardware Trojan environment. Based on this experimental observation, we undertook mathematical analysis to determine the relationship that could enhance sensor accuracy defined by the interdependence of temperature, threshold voltage, oscillation count/frequency, and ageing/delays variability.

As the DSS RO pairs are subjected to temperature and threshold voltage variations at time t, the oscillation count/frequency f_{tDSS} begins to fall. It becomes lower than the frequency $f_{0\,DSS}$ at time 0. This frequency degradation δf can, then, be given as:

$$\delta f = f_{0DSS} - f_{tDSS} \tag{13}$$

With the application of negative bias at three different values in time 0, the percentage frequency difference is resultantly calculated as:

$$\partial f_{0DSS} = \frac{f_{0DSS,V_{DD1}} - f_{0DSS,V_{DD2}} - f_{0DSS,V_{DD3}}}{f_{0DSS,V_{DD3}}} \tag{14}$$

where, $V_{DD1} > V_{DD2} > V_{DD3}$. As there exists a positive correlation between δf and ∂f_{0DSS}, we aim at identifying DSS RO pair that experiences a maximum frequency degradation relative to percentage frequency differences at the afore-mentioned negative bias and temperature stress values, mathematically:

$$\delta f \xleftarrow{\rho} \partial f_{0DSS} \tag{15}$$

Then, the frequency degradation for the sensor can be expressed as follows:

$$\delta f = \Delta f_t - \Delta f_0 \tag{16}$$

where,

$$\Delta f_t = f_{tFSS} - f_{tDSS} \tag{17}$$

We also consider the impact process variations (PVs) could have on frequency (delay/ageing) degradation δf and the percentage frequency difference ∂f_{DSS}. With the positive correlation between the two, it is possible to have an optimal estimate $\delta^{\Lambda} f$ for δf. Minimum mean-square error (MMSE) estimator, for instance, provides versatility to achieve reduced mean square error and make more realistic estimates [58]. The DSS RO degradation is, therefore, expressed using the minimum mean-square error (MMSE) estimator, as follows:

$$\delta \hat{f}_{DSS} = \rho \frac{\sigma_{\delta f_{DSS}}}{\sigma_{\partial f_{DSS}}} (\partial f_{DSS} - \mu \partial f_{DSS}) + \mu \delta f_{DSS} \tag{18}$$

where ρ defines the correlation between frequency degradation in dynamic sensor segment (δf_{DSS}) and percentage frequency difference (∂f_{DSS}); $\sigma_{\delta f_{DSS}}$ and $\sigma_{\partial f_{DSS}}$ connotate the standard deviations for δf_{DSS} and ∂f_{DSS} respectively. Whereas, $\mu \delta f_{DSS}$ and $\mu \partial f_{DSS}$ represent the mean for δf_{DSS} and ∂f_{DSS} respectively.

The MMSE estimator for the overall sensor degradation (δf), as opposed to a particular sensor segment, can now be expressed as follows:

$$\begin{aligned} \delta \hat{f}_s &= \Delta \hat{f}_t - \Delta \hat{f}_0 = \left(\hat{f}_{tFSS} - \hat{f}_{tDSS}\right) - \left(\hat{f}_{0FSS} - \hat{f}_{0DSS}\right) \\ &= -\left(\hat{f}_{0FSS} - \hat{f}_{tFSS}\right) + \left(\hat{f}_{0DSS} - \hat{f}_{tDSS}\right) \end{aligned} \tag{19}$$

Since, the frequency degradation is assumed to be negligible in case of fixed sensor segment RO pairs, $\hat{f}_{0\,FSS} = \hat{f}_{tFSS}$, the above equation can be written as:

$$\begin{aligned} &= \left(\hat{f}_{0DSS} - \hat{f}_{tDSS}\right) \\ &= \delta \hat{f}_{DSS} = \rho \frac{\sigma_{\delta f_{DSS}}}{\sigma_{\partial f_{DSS}}} (\partial f_{DSS} - \mu \partial f_{DSS}) + \mu \delta f_{DSS} \end{aligned} \tag{20}$$

The above relation implies that with ρ being positive, the higher percentage frequency difference between the FSS and DSS RO pairs will, in turn, maximise the sensor frequency *(and subsequent delay/ageing)* degradation. It is represented by the separation between two distributions at $t = 0$ and $t = t$. This further implies that in the sensor with more RO pairs to select from, the one with the maximum percentage frequency difference within DSS RO pairs at $t = 0$ must be selected. This results in maximising the distance between the two distributions of frequency difference and minimising the probability of false prediction, as shown in Fig. 15(a).

Keeping in view the above mathematical derivations and 'selection strategy'(*as delineated in process flow - Fig. 17*), the detectability of hardware Trojan by the sensor is set for optimisation. Accordingly, we define the process variations based on transistor length (L) and oxide thickness (Tox), as given in Table 5 and choose ***'PVc'*** class of process variations as an extreme (worst) case to determine the pre-trigger value of frequency degradation, relative to percentage shift in the threshold voltage. Also, the two sensor segments (FSS and DSS) are implemented close to each other to eliminate the impact of undefined environmental variations upon measurements and the accuracy of detection.

The process flow (Fig. 17) targets the selection of the best (with maximum frequency degradation) FSS and DSS

RO-pair by, initially, selecting all the six RO-pairs and then capturing their frequencies. These frequencies are stored by two vectors, defined as $\vec{f}_{FSS} = [f_{FSS1}, f_{FSS2}, f_{FSS3}]$ and $\vec{f}_{DSS} = [f_{DSS1}, f_{DSS2}, f_{DSS3}]$ and all the frequency differences are stored in a matrix defined as, $\Delta f = \left[\Delta f_{ij}\right] nxn$, where $\Delta f_{ij} = \vec{f}_{FSS}(i) - \vec{f}_{DSS}(j), \forall(i,j)$. If Δf_{ij} is positive, the fixed and dynamic RO-pair with minimum Δf_{ij} is selected. Otherwise, only negative Δf_{ij} values are taken to update Δf. In such a condition, the resulting distribution $g_0^{'}(.)$ presents a significantly reduced spread, as is evident in Fig. 15(c).

However, at time *t,* the distribution g_t (.) must be shifted to the right to increase δf even further. In such a condition, DSS RO is selected with maximum $\partial \vec{f} dss\,(j)$

$$= \frac{f_{0DSS,V_{DD1}}(j) f_{0DSS,V_{DD2}}(j) f_{0DSS,V_{DD3}}(j)}{f_{0DSS,V_{DD3}}(j)} \quad (21)$$

whereas, the corresponding FSS RO with maximum Δf_{ij} is selected to minimise the spread of both distributions, $g_0^{'}(.)$ and g_t (.). Once the optimal RO pair is selected, the frequency difference Δf_{ij} is then stored to form the distribution $g_0^{'}(.)$. The threshold frequency f_{th} is finally calculated, to be referred to for the detection of hardware Trojan by comparing it with the frequency differences of FSS and DSS RO segments implemented in FPGA under authentication.

REFERENCES

[1] S. F. Mossa, S. R. Hasan, and O. Elkeelany, "Hardware trojans in 3-D ICs due to NBTI effects and countermeasure," *Integration*, vol. 59, pp. 64–74, Sep. 2017.

[2] Y. Wang, L. Xu, Z. Yang, H. Xie, P. Jiang, J. Dai, W. Luo, Y. Yao, E. Hitz, R. Yang, B. Yang, and L. Hu, "High temperature thermal management with boron nitride nanosheets," *Nanoscale*, vol. 10, no. 1, pp. 167–173, Nov. 2017.

[3] E. A. Scott, J. T. Gaskins, S. W. King, and P. E. Hopkins, "Thermal conductivity and thermal boundary resistance of atomic layer deposited high-K dielectric aluminum oxide, hafnium oxide, and titanium oxide thin films on silicon," *APL Mater.*, vol. 6, no. 5, May 2018, Art. no. 058302.

[4] P. Mangalagiri, S. Bae, R. Krishnan, Y. Xie, and V. Narayanan, "Thermal-aware reliability analysis for Platform FPGAs," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design*, Nov. 2008, pp. 722–727.

[5] Y. Wang, H. Luo, K. He, R. Luo, H. Yang, and Y. Xie, "Temperature-aware NBTI modeling and the impact of standby leakage reduction techniques on circuit performance degradation," *IEEE Trans. Dependable Secure Comput.*, vol. 8, no. 5, pp. 756–769, Aug./Sep. 2011.

[6] T. Grasser, R. Entner, O. Triebl, H. Enichlmair, and R. Minixhofer, "TCAD modeling of negative bias temperature instability," in *Proc. Int. Conf. Simul. Semicond. Processes Devices*, Sep. 2006, pp. 330–333.

[7] A. Waksman and S. Sethumadhavan, "Silencing Hardware Backdoors," in *Proc. IEEE Symp. Security Privacy*, May 2011, pp. 49–63.

[8] B. Vaidyanathan, A. S. Oates, Y. Xie, and Y. Wang, "NBTI-aware statistical circuit delay assessment," in *Proc. 10th Int. Symp. Qual. Electron. Design*, no. 4, Mar. 2009, pp. 13–18.

[9] S. Khan and S. Hamdioui, "Temperature dependence of NBTI induced delay," in *Proc. IEEE 16th Int. On-Line Test. Symp.*, Jul. 2010, pp. 15–20.

[10] G. T. Becker, F. Regazzoni, C. Paar, and W. P. Burleson, "Stealthy dopant-level hardware Trojans: Extended version," *J. Cryptogr. Eng.*, vol. 4, no. 1, pp. 19–31, Apr. 2014.

[11] D. Patra, A. K. Reza, M. K. Hassan, M. Katoozi, E. H. Cannon, K. Roy, and Y. Cao, "Adaptive accelerated aging for 28 nm HKMG technology," *Microelectron. Rel.*, vol. 80, pp. 149–154, Jan. 2018.

[12] D. Agrawal, S. Baktir, D. Karakoyunlu, P. Rohatgi, and B. Sunar, "Trojan detection using IC fingerprinting," in *Proc. IEEE Symp. Security Privacy (SP)*, May 2007, pp. 296–310.

[13] J. Li and J. Lach, "At-speed delay characterization for IC authentication and Trojan Horse detection," in *Proc. IEEE Int. Workshop Hardware-Oriented Secur. Trust*, Jun. 2008, pp. 8–14.

[14] M. Abramovici and P. Bradley, "Integrated circuit security: New threats and solutions," in *Proc. 5th Annu. Workshop Cyber Secur. Inf. Intell. Res., Cyber Secur. Inf. Intell. Challenges Strategies (CSIIRW)*. New York, NY, USA: Association for Computing Machinery, 2009, pp. 1–3, Art. no. 55.

[15] R. S. Chakraborty, F. Wolff, S. Paul, C. Papachristou, and S. Bhunia, "*MERO*: A statistical approach for hardware trojan detection," in *Cryptographic Hardware and Embedded Systems—CHES 2009* (Lecture Notes in Computer Science), vol. 5747, C. Clavier and K. Gaj, Eds. Berlin, Germany: Springer, 2009.

[16] S. Narasimhan, D. Du, R. S. Chakraborty, S. Paul, F. Wolff, C. Papachristou, K. Roy, and S. Bhunia, "Multiple-parameter side-channel analysis: A non-invasive hardware Trojan detection approach," in *Proc. IEEE Int. Symp. Hardware-Oriented Security Trust (HOST)*, Jun. 2010, pp. 13–18.

[17] C. Lamech, R. M. Rad, M. Tehranipoor, and J. Plusquellic, "An experimental analysis of power and delay signal-to-noise requirements for detecting trojans and methods for achieving the required detection sensitivities," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 3, pp. 1170–1179, Sep. 2011.

[18] X. Zhang and M. Tehranipoor, "RON: An on-chip ring oscillator network for hardware Trojan detection," in *Proc. Design, Autom. Test Europe*, vol. 1, Mar. 2011, pp. 1–6.

[19] A. Ferraiuolo, X. Zhang, and M. Tehranipoor, "Experimental analysis of a ring oscillator network for hardware trojan detection in a 90 nm ASIC," in *Proc. Int. Conf. Comput.-Aided Design (ICCAD)*, vol. 12, 2012, p. 37.

[20] Y. Cao, C.-H. Chang, and S. Chen, "Cluster-based distributed active current timer for hardware Trojan detection," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, vol. 1, May 2013, pp. 1–4.

[21] O. Soll, T. Korak, M. Muehlberghuber, and M. Hutter, "EM-based detection of hardware trojans on FPGAs," in *Proc. IEEE Int. Symp. Hardware-Oriented Secur. Trust (HOST)*, May 2014, pp. 84–87.

[22] J. Balasch, B. Gierlichs, and I. Verbauwhede, "Electromagnetic circuit fingerprints for Hardware Trojan detection," in *Proc. IEEE Int. Symp. Electromagn. Compat. (EMC)*, Aug. 2015, pp. 246–251.

[23] X. T. Ngo, Z. Najm, S. Bhasin, S. Guilley, and J.-L. Danger, "Method taking into account process dispersion to detect hardware Trojan Horse by side-channel analysis," *J. Cryptogr. Eng.*, vol. 6, no. 3, pp. 239–247, Sep. 2016.

[24] P. Singh, E. Karl, D. Blaauw, and D. Sylvester, "Compact degradation sensors for monitoring NBTI and oxide degradation," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 20, no. 9, pp. 1645–1655, Sep. 2012.

[25] Y. Wang, M. Enachescu, S. D. Cotofana, and L. Fang, "Variation tolerant on-chip degradation sensors for dynamic reliability management systems," *Microelectron. Rel.*, vol. 52, nos. 9–10, pp. 1787–1791, Sep. 2012.

[26] Y. Wang, S. D. Cotofana, and L. Fang, "Statistical reliability analysis of NBTI impact on FinFET SRAMs and mitigation technique using independent-gate devices," in *Proc. IEEE/ACM Int. Symp. Nanosc. Archit. (NANOARCH)*, 2012, pp. 109–115.

[27] Y. Wang, S. D. Cotofana, and L. Fang, "Analysis of the impact of spatial and temporal variations on the stability of SRAM arrays and the mitigation technique using independent-gate devices," *J. Parallel Distrib. Comput.*, vol. 74, no. 6, pp. 2521–2529, Jun. 2014.

[28] S. V. Kumar, C. H. Kim, and S. S. Sapatnekar, "NBTI-aware synthesis of digital circuits," in *Proc. 44th ACM/IEEE Design Autom. Conf.*, Jun. 2007, pp. 370–375.

[29] A. Calimera, E. Macii, and M. Poncino, "Design techniques for NBTI-tolerant power-gating architectures," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 59, no. 4, pp. 249–253, Apr. 2012.

[30] Z. Abbas, M. Olivieri, U. Khalid, A. Ripp, and M. Pronath, "Optimal NBTI degradation and PVT variation resistant device sizing in a full adder cell," in *Proc. 4th Int. Conf. Rel., Inf. Technol. Optim. (ICRITO)*, Sep. 2015, pp. 1–6.

[31] I.-C. Lin, S.-M. Syu, and T.-Y. Ho, "NBTI tolerance and leakage reduction using gate sizing," *JETCJ. Emerg. Technol. Comput. Syst.*, vol. 11, no. 1, pp. 1–12, Oct. 2014.

[32] K.-C. Wu, D. Marculescu, M.-C. Lee, and S.-C. Chang, "Analysis and mitigation of NBTI-induced performance degradation for power-gated circuits," in *Proc. IEEE/ACM Int. Symp. Low Power Electron. Design*, Aug. 2011, pp. 139–144.

[33] W. H. Choi, H. Kim, and C. H. Kim, "Circuit techniques for mitigating short-term vth instability issues in successive approximation register (SAR) ADCs," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, Sep. 2015, pp. 1–4.

[34] S. Kiamehr, M. B. Tahoori, F. Firouzi, and M. Ebrahimi, "Extending standard cell library for aging mitigation," *IET Comput. Digit. Techn.*, vol. 9, no. 4, pp. 206–212, Jul. 2015.
[35] G. Zhang, M. Yi, Y. Miao, D. Xu, and H. Liang, "NBTI-induced circuit aging optimization by protectability-aware gate replacement technique," in *Proc. 16th Latin-Amer. Test Symp. (LATS)*, Mar. 2015, pp. 1–4.
[36] S. V. Kumar, C. H. Kim, and S. S. Sapatnekar, "Impact of NBTI on SRAM read stability and design for reliability," in *Proc. 7th Int. Symp. Quality Electron. Design (ISQED)*, Apr. 2006, p. 6.
[37] T.-H. Kim, R. Persaud, and C. H. Kim, "Silicon Odometer: An on-chip reliability monitor for measuring frequency degradation of digital circuits," *IEEE J. Solid-State Circuits*, vol. 43, no. 4, pp. 874–880, Apr. 2008.
[38] E. Saneyoshi, K. Nose, and M. Mizuno, "A precise-tracking NBTI-degradation monitor independent of NBTI recovery effect," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, Feb. 2010, pp. 192–193.
[39] X. Zhang and M. Tehranipoor, "Design of on-chip lightweight sensors for effective detection of recycled ICs," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 22, no. 5, pp. 1016–1029, May 2014.
[40] C. Dong, G. He, X. Liu, Y. Yang, and W. Guo, "A multi-layer hardware trojan protection framework for IoT chips," *IEEE Access*, vol. 7, pp. 23628–23639, 2019.
[41] S. Moein, T. A. Gulliver, F. Gebali, and A. Alkandari, "A new characterization of hardware trojans," *IEEE Access*, vol. 4, pp. 2721–2731, 2016.
[42] S. R. Hasan, B. Pontikakis, and Y. Savaria, "An all-digital skew-adaptive clock scheduling algorithm for heterogeneous multiprocessor systems on chips (MPSoCs)," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2009, pp. 2501–2504.
[43] S. R. Hasan, S. F. Mossa, O. S. A. Elkeelany, and F. Awwad, "Tenacious hardware trojans due to high temperature in middle tiers of 3-D ICs," in *Proc. IEEE 58th Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Aug. 2015, pp. 1–4.
[44] A. P. Shah, N. Yadav, A. Beohar, and S. K. Vishvakarma, "SUBHDIP: Process variations tolerant subthreshold Darlington pair-based NBTI sensor circuit," *IET Comput. Digit. Techn.*, vol. 13, no. 3, pp. 243–249, May 2019.
[45] A. Amouri, F. Bruguier, S. Kiamehr, P. Benoit, L. Torres, and M. Tahoori, "Aging effects in FPGAs: An experimental analysis," in *Proc. 24th Int. Conf. Field Program. Logic Appl. (FPL)*, Sep. 2014, pp. 5–8.
[46] X. Zhang, N. Tuzzio, and M. Tehranipoor, "Identification of recovered ICs using fingerprints from a light-weight on-chip sensor," in *Proc. 49th Annu. Design Autom. Conf. (DAC)*, 2012, pp. 703–708.
[47] U. Guin, D. Forte, and M. Tehranipoor, "Design of accurate low-cost on-chip structures for protecting integrated circuits against recycling," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 24, no. 4, pp. 1233–1246, Apr. 2016.
[48] W. Wang, Z. Wei, S. Yang, and Y. Cao, "An efficient method to identify critical gates under circuit aging," in *Proc. IEEE/ACM Int. Conf. Computer-Aided Design*, Nov. 2007, pp. 735–740.
[49] G. Wu, G. W. Deptuch, J. R. Hoff, and P. Gui, "Degradations of threshold voltage, mobility, and drain current and the dependence on transistor geometry for stressing at 77 K and 300 K," *IEEE Trans. Device Mater. Rel.*, vol. 14, no. 1, pp. 477–483, Mar. 2014.
[50] M. Omana, D. Rossi, N. Bosio, and C. Metra, "Low cost NBTI degradation detection and masking approaches," *IEEE Trans. Comput.*, vol. 62, no. 3, pp. 496–509, Mar. 2013.
[51] X. Wang, L. Winemberg, D. Su, D. Tran, S. George, N. Ahmed, S. Palosh, A. Dobin, and M. Tehranipoor, "Aging adaption in integrated circuits using a novel built-in sensor," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 34, no. 1, pp. 109–121, Jan. 2015.
[52] K. A. Bowman, J. W. Tschanz, N. S. Kim, J. C. Lee, C. B. Wilkerson, S.-L.-L. Lu, T. Karnik, and V. K. De, "Energy-efficient and metastability-immune resilient circuits for dynamic variation tolerance," *IEEE J. Solid-State Circuits*, vol. 44, no. 1, pp. 49–63, Jan. 2009.
[53] J. C. Vazquez, V. Champac, A. M. Ziesemer, R. Reis, J. Semiao, I. C. Teixeira, M. B. Santos, and J. P. Teixeira, "Predictive error detection by on-line aging monitoring," in *Proc. IEEE 16th Int. On-Line Test. Symp.*, Jul. 2010, pp. 9–14.
[54] E. Mintarno, V. Chandra, D. Pietromonaco, R. Aitken, and R. W. Dutton, "Workload dependent NBTI and PBTI analysis for a sub-45nm commercial microprocessor," in *Proc. IEEE Int. Rel. Phys. Symp. (IRPS)*, Apr. 2013, pp. 3A.1.1–3A.1.6.
[55] Y. Cao, "Cross-layer modeling and simulation of circuit reliability," *IEEE Trans. Comput. Des. Integr. Circuits Syst.*, vol. 33, no. 1, pp. 8–23, Jan. 2014.
[56] H. Kukner, M. Khatib, S. Morrison, P. Weckx, P. Raghavan, B. Kaczer, F. Catthoor, L. Van Der Perre, R. Lauwereins, and G. Groeseneken, "Degradation analysis of datapath logic subblocks under NBTI aging in FinFET technology," in *Proc. 15th Int. Symp. Quality Electron. Design*, Mar. 2014, pp. 1–7.
[57] G.-H. Lian, W.-Y. Chen, and S.-Y. Huang, "Cloud-based online ageing monitoring for IoT devices," *IEEE Access*, vol. 7, pp. 135964–135971, 2019.
[58] J. Tong, J. Yang, J. Xi, Y. Yu, and P. O. Ogunbona, "Tuning the parameters for precision matrix estimation using regression analysis," *IEEE Access*, vol. 7, pp. 90585–90596, 2019.

SOHAIB ASLAM received the B.E. degree in electrical engineering from the NED University of Engineering and Technology, Karachi, Pakistan, in 1997, and the M.Sc. degree in electronics and electrical engineering and management from the University of Glasgow, U.K., in 2014. He is currently pursuing the Ph.D. degree in manufacturing engineering (micro and nano electronics) with the Integrated Vehicle Health Management (IVHM) Centre, School of Aerospace, Transport, and Manufacturing (SATM), Cranfield University, U.K.

From 1997 to 2019, he served in Navy, MoD, Pakistan, as a Systems Engineer in electronic warfare. He has a wide-ranging experience of predictive and condition-based maintenance of naval electronic warfare systems. His research interests include semiconductor devices reliability and security assessment, mixed signals' sensor designing, and design for testability, prognostics, and security in field programmable gate arrays (FPGAs) and systems-on-chip (SoC).

IAN K. JENNIONS received the degree in mechanical engineering and the Ph.D. degree in CFD from Imperial College, London. He has worked for Rolls-Royce (twice), General Electric, and Alstom in a number of technical roles, gaining experience in aerodynamics, heat transfer, fluid systems, mechanical design, combustion, services, and IVHM. In July 2008, he moved to Cranfield University as a Professor and the Director of the IVHM Centre which is funded by a number of industrial companies, including Boeing, BAE Systems, Thales, Meggitt, MOD, DRS, Alstom Transport, and Novartis. He has led the development and growth of the IVHM Centre, in research and education, since its inception. His career spans some 40 years, working mostly for a variety of gas turbine companies. He has coauthored the book *No Fault Found—The Search for the Root Cause*.

Dr. Ian is a Fellow of IMechE, RAeS, ASME and PHM, a Contributing Member of the HM-1 IVHM Committee, a Chair of the SAE IVHM Steering Group, represents the Editorial Board of the *International Journal of Condition Monitoring*, the Director of the PHM Society, and a Chartered Engineer. He is the Editor of five SAE books on IVHM and the recent *The World of Civil Aerospace*.

MOHAMMAD SAMIE received the B.Sc. degree in electronics from the Azad University of Saveh, Iran, in 1997, the M.Sc. degree in electronics from Shiraz University, Shiraz, Iran, in 2002, and the Ph.D. degree in advanced electronics from the University of West of England, Bristol, U.K., in 2012. He is currently working as a Lecturer with the School of Aerospace, Transport and Manufacturing (SATM), Cranfield University, U.K. He is also leading Seretonix, a Secure and Reliable Electronic Systems Group, Cranfield University, with a focus on resilience and security of electronics. He has accumulated a wide and varied experience in field programmable gate arrays (FPGAs) and ASIC design, simulation, verification, and implementation (Toumaz in Didcot, U.K).

Dr. Samie was involved with two EPSRC-funded projects—NFF and SABRE, where he was responsible for creating most of the detailed designs and implementations. He has published 35 international journals, conference papers, and book chapters, with two awarded as best articles, on bio-inspired electronics.

SURESH PERINPANAYAGAM received the master's degree in engineering and the Ph.D. degree in engineering from Imperial College, London. He leads the ePHM Group, part of the Boeing Integrated Vehicle Health Management Research Centre set up by The Boeing Company. He has obtained grants amounting to £2M in total from industrial, EPSRC, Innovate U.K., and EU projects. He has spent considerable time in the industry working on various industrial Research and Development projects. His vision is to create every electronic system with its own brain to be self-aware of its own health state and to work effectively with other systems to complete a function even if it is not in an optimal state. To realize this, the group develops tools to detect the inception of failures in electronic components and track them to system failures. They also need to correlate the fundamental physics-of-failure work currently done at material science level (e.g., solder joint and wire bond failure) to the electronic system data acquired for system health management from data-centric aircrafts, such as Boeing 787. These intelligent electronic systems will inform and reconfigure its health state at different stages of its life. These new systems will redefine the current Built-In Test (BIT) technology in electronic systems with more user-friendliness, reduced no-fault-found rate, reduced repair, reduced cost, predictable failures, and greater availability for the industry.

Dr. Suresh is part of the working group developing the IEEE Standard for Prognostics and Health Management. He is also an Associate Editor of the IEEE TRANSACTIONS ON POWER ELECTRONICS Special Issue on Robust Design and Reliability in Power Electronics.

YISEN FANG received the B.Eng. degree in pharmaceutical engineering from Guangdong Pharmaceutical University, China, in 2013, and the M.Sc. degree in engineering and management of manufacturing systems from Cranfield University, U.K., in 2019.

From 2013 to 2018, he led a number of research projects with the Research and Development Department, Guangzhou Baiyunshan Tianxin Pharmaceutical Co., Ltd., China, that were focused on optimizing manufacturing plants and laboratory processes from safety and reliability perspective. His research interests include embedded electronics performance modeling, hardware reliability and security in manufacturing control systems, and augmented reality (AR) applications.

Mr. Yisen received second place with £1,000 in Unilever Ice Cream Innovation Accelerator Competition held in Cranfield University, U.K., in 2019.

2020-02-11

Ingress of threshold voltage-triggered hardware trojan in the modern FPGA þÿ f a b r i c d e t e c t i o n m e t h o d o l o g y

Aslam, Sohaib

IEEE

Aslam S, Jennions IK, Samie M, et al., (2020) Ingress of threshold voltage-triggered hardware þÿ t r o j a n i n t h e m o d e r n F P G A f a b r i c d e t e c t i o n m e t h o d o l o g y a n d m i t i g a t i o n. 8, 2020, pp. 31371-31397

https://doi.org/10.1109/ACCESS.2020.2973260